

# C64 SPECIAL COMMODORE

L'ORDINATEUR INDIVIDUEL HORS SERIE - N° 70 - 1985

CONNAITRE  
COMPRENDRE  
EXPLOITER  
**LE COMMODORE 64**



**BASIC-LISTE :**  
**1<sup>er</sup> ROMAN PROGRAMME**  
**POUR MEMOIRE VIVE**

**LUTINS**  
*Jeux, poignées,  
gammes et collisions*  
**LANGAGES**  
*Assembleur - Les deux Pascal*  
**PRATIQUE**  
*Disquettes : les premiers soins  
Pirates : halte au vol  
Mémoire : toutes les adresses*







# EDITORIAL

## MILLE IDEES POUR UN ORDINATEUR ADULTE

*Nom : Commodore 64. Profession : Ordinateur. Date de naissance : 1982. Date d'arrivée en France : 1983... Et nous sommes en 1985. Trois ans pour un ordinateur, c'est vieux, même très vieux. Le C. 64 a déjà fait couler beaucoup d'encre, alors que ses successeurs se font attendre...*

*Si nous lui dédions aujourd'hui – et aujourd'hui seulement ! – ce hors-série, c'est qu'un bon cru ne vieillit pas. Le temps permet d'en déguster les mille nuances et de le savourer avec la volupté du connaisseur. Vous trouverez dans ces pages une pluie de météorites, un déluge de conseils à piocher, d'idées à exploiter. En somme, une véritable mine que nous avons voulue accessible à tous – débutants et/ou initiés – pour un ordinateur que des dizaines d'ouvrages n'ont pas épuisé.*

*En cadeau : un super-roman noir pour un super-programme inédit, le premier du genre de la presse informatique. En suivant les aventures de Bernie Soap, Tony Jennett et Mimi Flytox à la poursuite du Basic-Liste, vous en saurez plus sur l'anatomie du Commodore 64 et de ses ancêtres sans mourir d'ennui.*

*Curieux : à vos claviers ! Nous avons choisi d'exciter votre intellect, de stimuler vos neurones. Ceux qui veulent que « ça marche tout seul » trouveront quelques programmes à consommer, mais nous avons volontairement minimisé la place réservée aux jeux et aux logiciels « dits » professionnels, quelle que soit par ailleurs la grande qualité des uns ou des autres. Fouineurs : vous êtes privilégiés ! Nous vous proposons une foule de trucs pratiques et de multiples références aux articles déjà publiés dans L'Ordinateur Individuel, nos amis de List et de la Commode. Ce numéro posera peut-être à certains plus de questions qu'il n'en résout, mais il leur donnera les moyens d'en trouver les réponses. Nous l'avons choisi ainsi pour ne pas gêner votre plaisir ni brider votre imagination. A vous l'aventure.*

LA REDACTION ■

## L'ORDINATEUR INDIVIDUEL

Editeur : Jean-Pierre Nizard.  
Directeur de la rédaction : Bernard Savonet.

### REDACTION

Directeur de la rédaction : Bernard Savonet.  
Antoine Jennet (rédacteur en chef délégué).  
Martine Solirene (secrétaire générale de rédaction).  
Rédaction : Jean-Luc Goudeat, Michel Arditti.  
Fabrication :  
Mireille Othnin-Girard (secrétaire de rédaction).  
Viviane Bazin (révision).

### PUBLICITÉ-PROMOTION-VENTES

Editeur adjoint : Jean-Daniel Belfond.  
Chefs de publicité : Marie-Christine Seznec et Philippe Daniloff assistés de Fatma Bouilla et de Marie-Christine Jugeau.  
Administration : Maryse Marti assistée de Anne Stolkowski.  
Promotion : Brigitte Millé assistée de Geneviève Cuvellier.  
Ventes-Diffusion NMPP : Béatrice Ginoux-Defermon.

### Coordination

Patrick Brai  
Conseillers techniques  
François-Jean Bayard, Daniel-Jean David  
Conception graphique, secrétariat de rédaction,  
maquettes  
Thierry Othnin-Girard  
Correction  
Simone Damotte

### RÉDACTION, VENTE, PUBLICITÉ

France et étranger  
5, place du Colonel-Fabien, 75491 Paris Cedex 10.  
Tél. (1) 240 22 01. Téléx : 215 105F LORDI  
Belgique :  
Rédaction et publicité : 3, av. de la Ferme-Rose, B-1180 Bruxelles.  
Tél. (2) 345 99 10.  
Vente et abonnement : Soumillon, 28, av. Massenet,  
B-1180 Bruxelles. Tél. (2) 344 19 23.  
Suisse :  
19, route du Grand-Mont, CH-1052 Le Mont-sur-Lausanne. Tél.  
(21) 32 15 65.

La loi du 11 mars 1957 n'autorise aux termes des alinéas 2 et 3 de l'article 41 d'une part, que « les copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective », et d'autre part, que les analyses et les courtes citations dans un but d'exemples et d'illustrations, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants-droit ou ayants-cause est



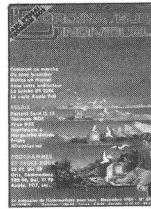
# 4 NUMEROS DE L'OI POUR 70 FRANCS



**62 LES OUTILS POUR MIEUX PROGRAMMER :** assembleurs, éditeurs, débogueurs, renumérotation, aides au dessin, à la conception musicale • Coup d'œil sur : HP 71 B, GP 50, Quick Disk 2, Omnis 1.2.3 • Essais : Macintosh ; Lotus 1.2.3, Flight Simulator ; logiciels de jeux • Panorama des machines MSX • 8, 16, 32 bits : comment s'y reconnaître • Programmation et jeux : Puissance 4 (II), calendrier lunaire sur Apple 2, jeu de rôles sur PC 1212 et 1251, écrire hébreu, cyrillique ou chinois sur TI 99/4A, dessin sur Vic 20 ; à la recherche de l'octet perdu (2) • Festival du logiciel, compte rendu • Expos : SICOB 84, International Business Show et Micro Computer Show à Tokyo, NCC à Las Vegas.



**63 L'ORDINATEUR DE LA PRIMAIRE A LA FAC :** l'enseignement assisté par ordinateur (EAO) ; LSE ; 100 000 micros dans les écoles : l'informatique dans les écoles de commerce et à Polytechnique • Coup d'œil sur : IBM PC/AT, Magnum, Amstrad, MTX 512, Laser 3000 • Essais : PAP, HP 110 ; Mu-math, Basic sans peine ; logiciels de jeux • Débutants : les réponses à vos questions • Programmation et jeux : Puissance 4 (III), dans les dédales de la planète Protos sur Oric, faire chanter son ZX 81, procédures récursives sur Pet/CBM, adresses des routines existant dans les moniteurs ; à la recherche de l'octet perdu (3) • Le petit assembleur illustré (1) ; langage C : les structures de données.



**64 COMMUNICATION :** réseaux, modems. Comment ça marche, où vous brancher ? Mettez un Minitel dans votre ordinateur. La liaison RS 232C. La carte Apple Tell à l'essai • Coup d'œil sur : Yeno DPC 64, Sord IS 11, imprimante Uchida DWX 305 • Essais : MO 5 ; Friday, Microplanner ; logiciels de jeux • Débutants : le fichier d'adresses (L'OI 59) d'un ordinateur à l'autre • Programmation et jeux : contrôler un modem sur DAI, morpion sur ZX 81, missiles sur VIC 20, transformer son Oric en périodmètre, nombres premiers sur Assembleur Z 80, labyrinthe sur TRS 80 ; solution des jeux n° 493 et 494 ; à la recherche de l'octet perdu (4) • Expo : Sicob, les nouveautés • Le petit assembleur illustré (II) ; langage C : interactions avec son environnement.



**66 CHOISIR SON IMPRIMANTE :** 141 imprimantes de 1000 à 15 000 F et leurs caractéristiques techniques ; faire le bon choix. • Coup d'œil sur : PC AT, les ordinateurs portatifs, VG 5000, Spectrum. • Essais : Amstrad CP 464, Yeno SC 3000 ; Habadex, Edi Logo. • Programmation et jeux : au-delà de la 3<sup>e</sup> dimension sur Apple 2, calcul mental sur Canon XO 7, plonger dans la mémoire de tout ordinateur, haute finance sur PC 1212, murailles de briques sur TI 99/4A, traitement de texte sur Apple. P.a.i. : les chaînes et l'affichage. • Expos : Comdex de Las Vegas et d'Amsterdam. • Index des trucs et astuces.

**65 LANGAGES : DÉPASSEZ VOTRE BASIC.** Panorama des langages. Méthode d'écriture de programmes. Langages disponibles sur votre OI. • Coup d'œil sur : Commodore Plus 4, One, Einstein, Ericsson PC. • Essais : Apple 2C ; Publi-Base, File Vision. • Programmation et jeux : au centre d'une mémoire vive sur New Brain, double précision sur TRS 80 ou Vidéo Génie, désassembler avec PC 1500, transmission de pensées sur Commodore, sauvegarde sur DAI (I), documentation sur Apple. P.a.i. : pour ne pas sortir de la routine. • Langage C : gérer un ensemble de messages. • Microsoft et les standards. L'autocritique d'IBM. Informatique au Cap d'Agde. Langage idéal. • OI et Minitel. • Répertoire 83-84.



**67 LES PÉRIPHÉRIQUES,** ceux qui vont plus loin, du modem à l'interface secteur ; Le CX 5M synthétiseur ; Images sur DAI et IBM PC ; La carte porte-parole sur Apple 2e ; Le bras robot de Multisoft. • Coup d'œil sur : OPD d'ICL. • Essais : Epson PX 8, Premium Softcard ; Thinktank, Rédacteur. • Programmation et jeux : graphique en 3 dimensions sur Canon XO7, tours de Hanoi sur TI 99/4A ou New Brain, meurs d'un interpréteur Basic ; SV 318, courbes sur Apple, brancher une chaîne hifi sur CBM 64 et VIC 20, le triomphe de l'intelligence artificielle. P.a.i. : la routine idéale. • Créature au Carrefour de la Communication ; l'avenir de l'informatique d'après B. Lussato ; Wang ; le piratage ; les disquettes de démonstration.



**68 PANORAMA PRINTEMPS 85 :** plus de 200 ordinateurs comparés, tous les derniers-nés, répertoire des constructeurs et importateurs. • Coup d'œil sur : Intégral PC, Hector MX, Lys, Commodore 16, Guépard, Alice 90 ; Wordstar 2000. • Essais : HP 150 • Programmation et jeux : Rubrique CP/M ; Stat.com, multiplication sur FX 702 P, Basiciso sur CBM 64, morse sur PC 1212 ; P.a.i. : le programme de vos chaînes. • Expo : Consumer Electronic Show à Las Vegas.



# SOMMAIRE

<b>ÉDITORIAL.</b> <i>Mille et une idées pour un ordinateur adulte</i>	<b>3</b>
<b>RUN.</b> <i>Je viens de recevoir mon C. 64</i>	<b>6</b>
<b>PÉRIFÉRIQUES.</b> <i>Cassettes, gaufrettes, disquettes, et toute la bande</i>	<b>8</b>
<b>SAUVEGARDE.</b> <i>Protégez vos programmes</i>	<b>11</b>
<b>ASSEMBLEUR.</b> <i>Les objets manipulables en L.M.</i>	<b>13</b>
<b>MANETTES.</b> <i>Lutins, poignées et collisions</i>	<b>16</b>
<b>GRAPHIQUES.</b> <i>Boîte à outils</i>	<b>21</b>
<b>ALENVER.</b> <i>Un programme renversant</i>	<b>23</b>

Premier roman-programme en série noire



## ENQUETE BASIC-LISTE

I. <i>Autopsie d'une mémoire vive</i>	<b>25</b>
II. <i>Tony Jennett change d'adresses</i>	<b>30</b>
III. <i>Le programme « Basic-liste »</i>	<b>32</b>
IV. <i>Vidéo renversée pour assembleur simulé</i>	<b>39</b>

<b>PASCAL.</b> <i>Le match Zoom-Oxford</i>	<b>46</b>
<b>COPIE.</b> <i>La H.R. sur imprimante MPS 801</i>	<b>47</b>
<b>MODEMS.</b> <i>RS 232 et la fin de l'île déserte</i>	<b>50</b>
<b>GAMMES.</b> <i>SID chip : la puce mélomane</i>	<b>52</b>
<b>RÉFLEXES.</b> <i>Jeux : distraire son Commodore</i>	<b>55</b>
<b>CARTE MÉMOIRE.</b> <i>Les adresses stratégiques</i>	<b>58</b>
<b>TRUCASTUCES.</b> <i>La foire aux programmes</i>	<b>63</b>
<b>LIVRES.</b> <i>Que sais-je ?</i>	<b>64</b>



# RUN

## JE VIENS DE RECEVOIR MON C. 64

**Vous venez de recevoir votre C. 64, c'est votre (presque) premier ordinateur. Préparez-vous de belles heures d'exploration et d'art pur avec les trois circuits-vedette du C. 64 : SID (Sound Interface Device, un générateur de son supérieur à celui qui équipe certains synthétiseurs), VIC II (Vidéo Interface Chip, deuxième du nom, à qui le VIC-20 devait le sien, bien sûr, mais à qui le deux en chiffres romains s'il vous plaît confère une distinction royale) et 6510, le microprocesseur central, compatible avec le 6502 qui a fait la gloire du PET, de l'Apple, de l'Oric et autres Atari.**

**D**ans la fébrilité qui précède ou qui suit l'achat, vous avez peur de faire un branchement fatal, faute du cordon ou de la prise adéquat ? Rassurez-vous, c'est virtuellement impossible si vous vous en tenez au matériel contenu dans son « bel » emballage : le 64 lui-même, le boîtier d'alimentation secteur, un cordon télévision (parfois fourni à part), un manuel d'utilisation. Si vous n'avez que la version anglaise ou allemande, votre C. 64 a été emporté par un circuit *parallèle*. Exigez quand même une traduc-

tion, surtout en cas d'achat par petites annonces.  
 • Première connexion : l'alimentation. La fiche DIN ronde se branche sur le côté droit, près de l'interrupteur, la fiche secteur dans la prise électrique de votre mur. Testez l'interrupteur : le témoin sur le dessus du 64 doit s'allumer. Sinon, avant de rapporter le tout au vendeur, assurez-vous quand même qu'il y a du *jus* dans votre prise murale, tous les manuels vous le diront.

D'une manière générale, n'essayez pas de brancher ou de débrancher quelque prise que ce soit pendant que votre C. 64 est sous tension : il ne se produira rien de méchant jusqu'au jour où... Pour le reste, regardez la face arrière. Les prises sont destinées, de gauche à droite, aux cartouches (la grande plate), à l'antenne télévision, à la télévision encore, à l'imprimante et/ou à l'unité de disquettes, au magnétophone (la petite plate), la dernière est le port utilisateur.

Deux prises pour la télévision ? Oui, mais une à la fois. La petite ronde est à relier à la prise d'antenne du téléviseur, l'autre, une DIN cinq broches peut recevoir un cordon pour l'entrée PERITEL. Laquelle prendre ? Cela dépend. Trois versions du C. 64 sont actuellement disponibles en France :

– **PAL** : la plus facile à trouver (importations *parallèles*). Ne peut se connecter par prise d'antenne que sur un téléviseur PAL ou bistandard. En PERITEL, on peut aussi se brancher sur un téléviseur SECAM, mais l'image s'inscrit en noir et blanc.

### SID

■ Vous cherchez le son qui fera choir les producteurs sur leurs financières fesses ? SID le permet. Pour musiciens et programmeurs avertis qui ne se contentent pas du haut-parleur de la télé.

CARTOUCHES D'EXTENSIONS

JACK  
TÉLÉVISION

TÉLÉ



– **SECAM** : Cocorico ! Connexion antenne ou PERITEL sans problème sur un téléviseur français.  
 – **RVB** : La version la plus récente. Se connecte par prise PERITEL aux téléviseurs PAL ou SECAM, ainsi qu'aux moniteurs vidéo. Attention donc à choisir la version du 64 qui convient à votre téléviseur.

#### Un bon conseil :

Si votre téléviseur est (trans)portable, emportez-le chez votre revendeur. Il se chargera du branchement et des réglages, et vous serez certain de la compatibilité des appareils, et des cordons. Cela est valable pour tous les périphériques.

Vous avez maintenant, à la mise sous tension, le message d'entrée COMMODORE BASIC, le READY et un curseur clignotant ; vous allez frapper votre premier programme et vous voudrez le sauvegarder. La prise plate du magnétophone étant asymétrique, il n'est pas possible de la connecter à l'envers. Quant à l'unité de disquettes et à l'imprimante, si vous êtes effrayé par la présence d'une seule prise pour les deux, celle que la documentation Commodore appelle *interface IEEE série*, rassurez-vous, chaque appareil comporte une prise qui permet de brancher le suivant, selon un assemblage en chaîne.

Le connecteur de gauche est destiné aux cartouches, un des bons moyens de charger instantanément des jeux, des utilitaires ou des programmes « sérieux », mais une fatigue pour les contacts si l'on veut en changer souvent. A ne brancher ou débrancher que sur C. 64 éteint.

### L'aventure aux passionnés

La grande aventure informatique est commencée. Qui peut savoir où elle vous mènera, tant il est vrai que les possesseurs de Commodore 64 sont d'un lyrisme grandiose doublé d'une mau-

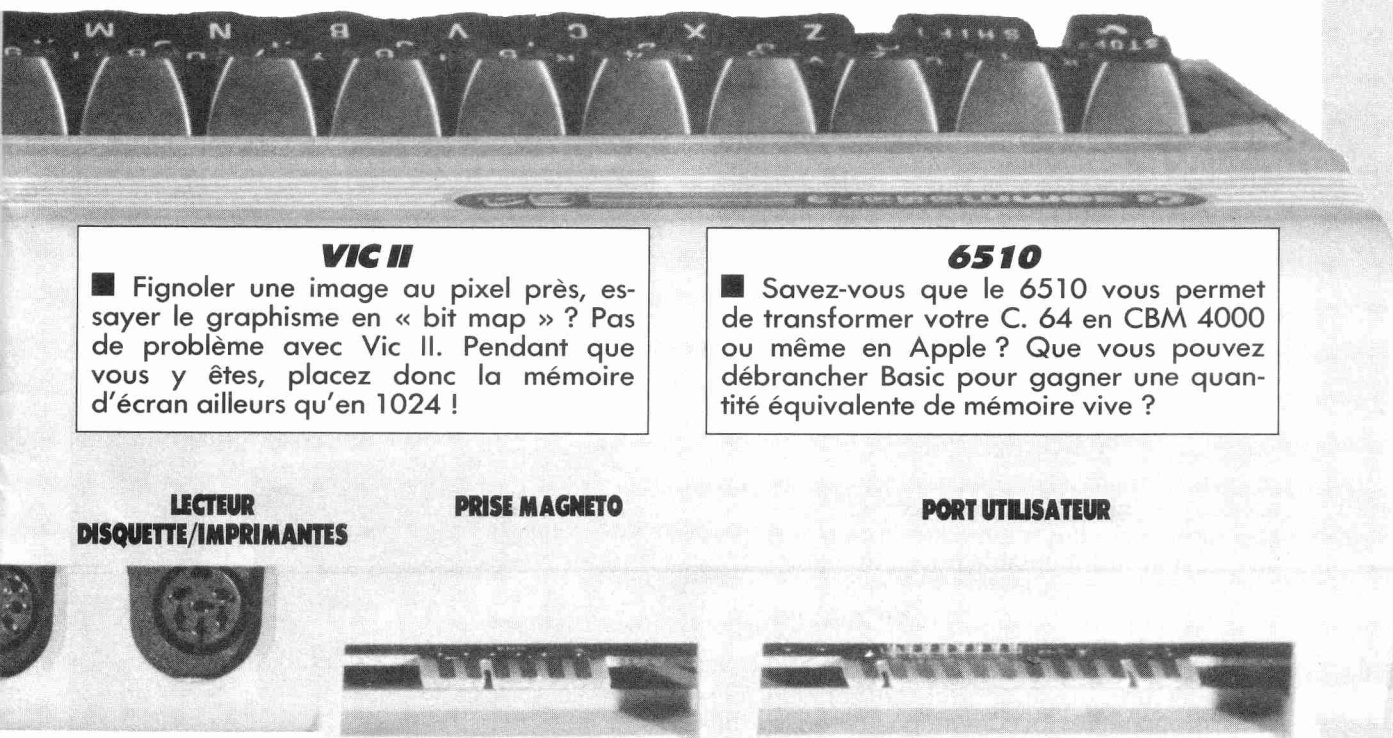
vaise fois éhontée lorsqu'il s'agit de vanter les mérites de leur bécane. On croirait entendre des parents parler de leur gosse ; soit il a toutes les qualités (détendu-intelligent-équilibré), soit il est plein de défauts, mais malheur à qui oserait approuver et en citer un seul !

C'est vrai, on commence par râler lorsqu'après avoir cru qu'on aurait 64 Kilo-octets de mémoire pour ses programmes en Basic, on s'aperçoit qu'on est loin du compte. On râle parce que l'alimentation chauffe sournoisement les orteils, on râle parce que la disquette lambine, on râle parce que les copains ont des PAINT, des DRAW, des CIRCLE, des BEEP et des SOUND à ne pas savoir qu'en faire au bout de leur Basic, on râle par principe, par habitude et par amour.

Mais quand on entend chez les mêmes copains des musiquettes nasillardes et simplistes, quand on voit s'animer sur leur écran des personnages saccadés et mal dégrossis, quand on trouve sur leur machine une adaptation rudimentaire du jeu publié six mois plus tôt pour la sienne, on ne dit rien parce qu'on a le triomphe modeste. On a un petit sourire condescendant qui fait du bien par où il passe.

Il faut faire des acrobaties pour lui faire sortir une fugue du vieux J.S. Bach ou le dernier Michael Jackson ? Peut-être, mais C. 64 peut le faire. Et avec ses trois voix. Il faut se farcir quelques nuits blanches avant de comprendre ses Lutins, et un bon week-end pour les mettre en œuvre ? Peut-être, mais on ne perd pas son temps quand on apprend, et encore moins quand on comprend. Et puis, il ne manque pas de programmes et d'utilitaires pour simplifier la tâche de ceux à qui le temps manque. ■

François-Jean Bayard,  
d'après Pierre-Etienne Thalberg.



#### VIC II

■ Fignoler une image au pixel près, essayer le graphisme en « bit map » ? Pas de problème avec Vic II. Pendant que vous y êtes, placez donc la mémoire d'écran ailleurs qu'en 1024 !

#### 6510

■ Savez-vous que le 6510 vous permet de transformer votre C. 64 en CBM 4000 ou même en Apple ? Que vous pouvez débrancher Basic pour gagner une quantité équivalente de mémoire vive ?

LECTEUR  
DISQUETTE/IMPRIMANTES

PRISE MAGNETO

PORT UTILISATEUR



# PÉRIFÉRIQUES

## CASSETTES, GAUFRETTES, DISQUETTES IMPRIMANTES ET TOUTE LA BANDE

**La mémoire vive s'envole, la mémoire de masse reste, ainsi l'informatique transforme-t-elle les vieux proverbes. Pour ne pas perdre la mémoire, les ordinateurs font appel à des périphériques destinés à stocker et à reproduire programmes et données, dont la rapidité et l'efficacité sont, en général, égales au carré du prix !**

**A** tout seigneur tout honneur, le magnétophone à cassette est le plus courant. Celui du C. 64 lui est spécifique. Il est particulièrement lent, mais quelle fiabilité ! Et si par hasard vous écopez d'un LOAD ERROR irréductible, il vous reste encore une chance (voir le répertoire des Trucastuces dans ce numéro page 63).

Une mémoire de masse du troisième type commence à faire son apparition sur le marché français. Elle utilise un support nommé *wafer* (*gaufrette* en français), qui n'est autre qu'une bande sans fin montée dans une cassette spéciale. Intérêt de la chose : c'est le système d'exploitation qui se charge de repérer le passage voulu sur la bande.

Le « Quick Data Drive » se branche sans interface sur la prise magnétophone du C. 64. Il utilise des Microwafers (15 à 25 Ko), à système de protection par volet coulissant. Le système d'exploitation se charge en mémoire et offre les LOAD, SAVE et verify **habituels** à une vitesse environ dix fois plus rapide que sur cassette. Utilitaires fournis : catalogue détaillé d'un *wafer*, formatage, nettoyage (mais oui !) et copie de disque ou cassette à *wafer* si les programmes originaux ne sont pas protégés. Seul point noir, l'absence de logiciels édités sous ce format.

Le lecteur de disquette reste la mémoire de masse *noble*, celle qui permet d'avoir accès à une information précisément repérée sans avoir besoin de lire toutes les autres, celle indispensable pour des utilisations semi-professionnelles. En permettant de n'avoir en mémoire centrale que les informations nécessaires à un moment donné du programme, elle permet aussi au C. 64 de traiter une masse énorme de données (certains programmes occupent quatre ou six disquettes !).

Les disquettes, quant à elles, représentent beaucoup plus que le prix que vous les avez payées : elles représentent le prix des programmes et des données qui y sont inscrits, le temps ou l'argent consacré à leur acquisition. Sans un minimum de précautions, tout cette richesse risque de s'évanouir au gré d'une poussière ou d'un champ magnétique...

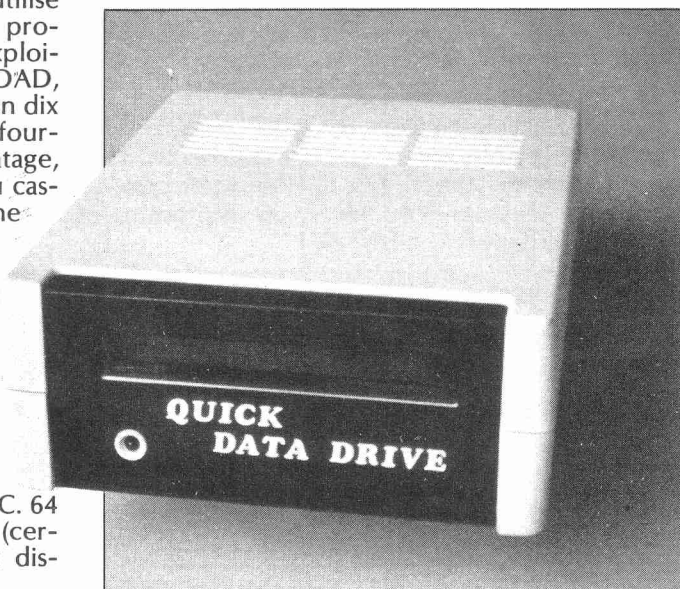
Il existe plusieurs types de lecteurs, et donc plusieurs types de disquettes, qui diffèrent par les caractéristiques suivantes :

– **La dimension** : 3, 5 1/4 ou 8 pouces (8, 13 ou 20 cm). Pour VIC 20 ou C. 64 : 13 cm (5 pouces 1/4).

– **Le nombre de faces utiles** : Le lecteur 1541 n'utilise qu'une seule face. Des disquettes simple face suffisent. Les doubles faces sont plus chères, mais fonctionnent parfaitement : qui peut le plus peut le moins !

– **La densité** : Dépend de la qualité du revêtement magnétique. VIC 20 et C. 64 se contentent de simple densité, mais, là encore, si l'on vous offre de la double densité au même prix...

– **La sectorisation** : Peut être matérielle ou logicielle (Hard-sectored ou Soft-sectored). Pour les reconnaître, faites tourner la disquette dans son enveloppe avec précaution et trois doigts passés dans le trou central. Si ce grand trou est entouré d'une couronne de petits, c'est une sectorisation



Plus rapide que le magnétophone, moins cher et moins performant que le lecteur de disquettes : le « Quick Data Drive » à « Microwafers ».



matérielle. S'il n'y a qu'un petit trou excentré, c'est une sectorisation logicielle. Les lecteurs de disquette Commodore fonctionnent sous sectorisation logicielle, mais ne se servent même pas du trou de synchronisation. N'importe quelle disquette 5 pouces 1/4 conviendra donc.

Un dernier point pourtant : il est préférable d'utiliser des disquettes ayant le trou central renforcé. Leur centrage dans la platine a alors toutes les chances d'être parfait et les chances d'usure du noyau sont moindres.

Il ne suffit pas de savoir quel type de disquettes acheter ; encore faut-il en prendre soin. Trois types de précautions s'imposent : les unes concernent le lecteur, d'autres concernent les disquettes elles-mêmes, d'autres enfin leur utilisation logique.

**Ne posez pas  
votre cigare sur le lecteur,  
mademoiselle...**

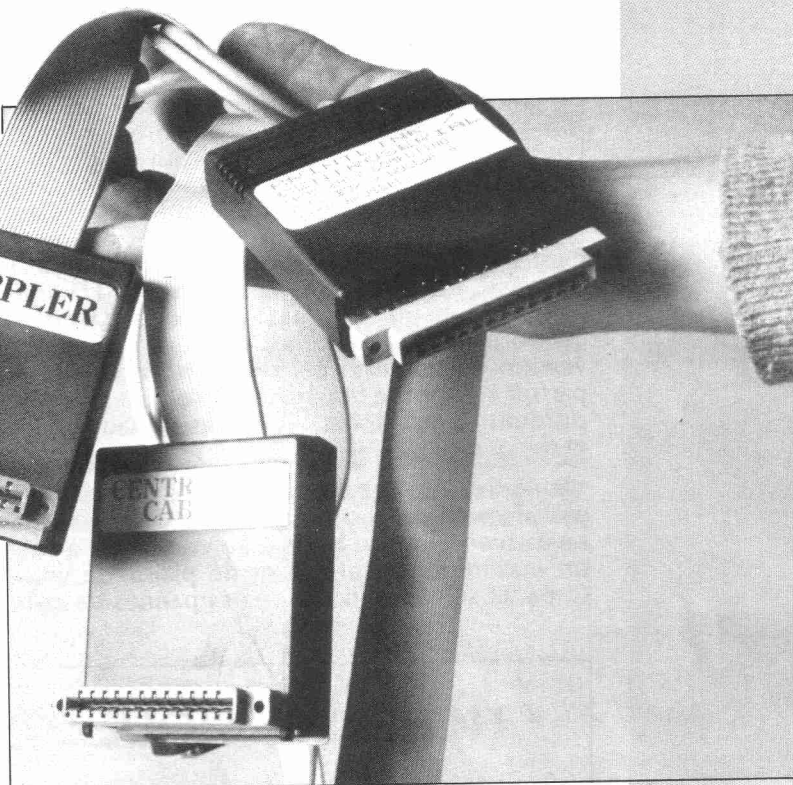
L'unité de disquettes doit être placée sur un support rigide, et ce pour deux raisons. La première est qu'elle ne doit subir ni chocs ni vibrations pendant les opérations de lecture ou d'écriture. La seconde est que tout échauffement lui est nuisible. Quel que soit votre degré d'intimité avec votre unité de disquettes, ne la posez donc pas sur votre couette ! Toutes les fentes de ventilation doivent être bien dégagées, aussi bien dessus que dessous. Il y a même des maniaques qui posent leur lecteur *sur le côté*. Les performances mécaniques ne risquent-elles pas d'en souffrir un peu ?

Les poussières aussi lui sont nuisibles : elles peuvent empêcher la bonne circulation de l'air ou se placer sur la tête de lecture. Enfin, il est important de prendre le plus grand soin des cordons d'alimentation et de raccordement à l'unité centrale ainsi que des connecteurs : un faux contact est si vite arrivé. Quant aux maniaques du tournevis qui bricolent du matériel sous garantie, ils doivent savoir qu'ils prennent des risques.

La surface magnétique des disquettes est particulièrement fragile, et la plupart des fabricants font figurer sur la pochette les principales consignes de sécurité : ne pas toucher la surface, ne pas laisser à la chaleur ou au soleil, éviter tout contact dur, crayon à bille par exemple (remplir l'étiquette *avant* de la coller ou utiliser un crayon feutre), ne pas chercher à sortir une disquette de son enveloppe, ne pas plier, ça va de soi, ne pas laisser traîner une disquette sans pochette et éviter les champs magnétiques. Il est enfin déconseillé de mettre l'unité de disquettes hors-tension ou sous tension s'il y a une disquette dedans.

Il n'est, hélas, pas toujours possible d'éviter ces accidents de manipulation, malgré toutes les précautions prises. A nous de nous débrouiller pour en réduire les conséquences.

On ne le dira jamais assez : faites des copies de vos disquettes importantes, une, deux, trois, selon l'importance de la disquette, avec de préférence un exemplaire dans une autre pièce.



Si vous connaissez quelqu'un qui possède une unité double CBM 4040, n'hésitez pas à lui demander de faire des copies de sécurité avec la commande BACKUP du BASIC 4.0. Une instruction et une minute suffisent pour reproduire une disquette vierge sur la disquette pleine ! Votre intérêt est de *toujours protéger la disquette-source en écriture* en recouvrant l'encoche prévue avec une étiquette. Gare aux commandes meurtrières comme NEW (formatage) ou SCRATCH (suppression) : leur emploi dans un moment d'inattention peut faire des ravages. Là encore, une étiquette de protection en écriture peut être utile, car on ne va pas l'enlever par inadvertance...

On peut aussi chercher à réduire les chances d'accident *au moment même de l'écriture d'un programme*. Le principe est celui-ci : ne pas laisser un fichier ouvert à un moment du programme où il y a risque de plantage. Car un fichier ouvert et non refermé est un fichier perdu, et fait même courir des risques à l'ensemble de la disquette. Une fausse manœuvre pendant la saisie des données, cela peut arriver ; l'astuce consiste à saisir d'abord toutes les données, puis, une fois qu'on est certain qu'elles sont entrées sans problèmes, à les transférer sur disquette, d'un seul coup d'un seul.

**L'imprévisible :  
c'est prévu !  
Merci Commodore**

Trois recommandations pour terminer vous éviteront peut-être des surprises avec les particularités Commodore :

- Un INPUT# ne peut pas lire d'enregistrements de plus de 80 caractères sous peine de plantage. Préférez GET# ou morcelez.
- L'opération *sauvegarde-et-remplacement* à l'aide de la commande "arobas" est l'objet d'une splendide erreur dans le système d'exploitation

Trippler : une des interfaces pour les liaisons parallèles avec votre imprimante. A essayer avant de l'adopter.



de la 1541 : de temps en temps, mais pas toujours, si l'on remplace une version d'un fichier par une version plus longue, ce qui est le cas la plupart du temps, on perd l'ancienne version du fichier, la nouvelle version du fichier, et, quand la 1541 est en forme, on perd aussi le catalogue, c'est-à-dire la disquette... Lorsqu'on fait un SAVE "arobas", le système sauvegarde d'abord la nouvelle version, puis modifie le catalogue : il efface les adresses (pistes et secteurs) de l'ancienne version, les remplace par celles de la nouvelle, et parfois se trompe d'adresse. Il n'en est que plus prudent de multiplier les copies de sauvegarde et de jouer du VERIFY.

Une telle collection de « Faites... Ne faites pas... » peut ne pas paraître aimable, mais vous savez bien que son but est de vous aider à tirer un maximum de service et de plaisir de votre C. 64. Et si le Destin frappe (les pannes de cou-

## FICHE TECHNIQUE

### INTERFACE PRINT LINK

Permet de faire fonctionner une imprimante parallèle avec le C.64.

Nécessité de charger un logiciel de transfert. Ne fonctionne pas avec tous les programmes du C. 64.

Importateur : Run Informatique  
Prix : 490 FF (K7), 540 FF (disquette).

### INTERFACE TRIPLER

Pour adaptation d'imprimante parallèle. Logiciel en MEM.

Ne permet pas de sortir les caractères graphiques du C. 64.

Importateur : Run Informatique  
Prix : 850 FF.

### INTERFACE CENTROSERIAL

Indispensable pour Brother EP44.

Importateur : Run Informatique  
Prix : 1 350 FF (interface + câble + logiciel).

### INTERFACE INTERPODE

Double norme : IEEE 488 et RS232C  
Permet l'adaptation des doubles drives série 4000 et 8000 et les autres périphériques professionnels Commodore, dont les capteurs de laboratoire.

Permet l'adaptation des périphériques standard RS232 (imprimantes, mais aussi modems, tablettes graphiques, etc.).

Importateur : Run Informatique.  
Prix : 1 300 FF (boîtier + alimentation + câble).

### COMMODORE VIC 1011A

Interface RS232C. Permet la connexion de modems et de tous les périphériques RS232.  
Prix : 450 FF.

### COMMODORE MPS 801

Imprimante matricielle - Traction 30 cps - 2,5 kg - 80 colonnes. IEEE Série.  
Prix : 2 400 FF.

## FICHE TECHNIQUE

### MAGNÉTOPHONE COMMODORE 1530

Importateur : Procep. Prix : 490 FF.

### UNITÉ DE DISQUETTES

#### COMMODORE 1541

Disquettes 5 pouces 1/4, simple face, simple densité soft-sectoré. Capacité : 170 Ko par face.

Importateur : Procep.

Microprocesseur 6502 intégré. Système d'exploitation en MEM.

Prix : 2 780 FF.

### QUICK DATA DRIVE

Alimentation incorporée.

Temps moyen d'accès :

Capacité	Temps
35 Ko	15 s
65 Ko	25 s
120 Ko	43 s

Importateur : Eurêka Informatique

Prix : 1 000 FF.

rant, ça arrive aux plus méticuleux), vous pourrez prendre la chose avec un éclat de rire ; vous sortirez du placard un autre exemplaire du programme en faisant la nique au Destin.

L'imprévisible : c'est prévu !

**La trace écrite  
Ça circule sur les  
périphériques !**

Pour garder une trace écrite des programmes et des fichiers, il faut passer par l'achat d'une imprimante. Pour le C. 64, c'est l'évidence, c'est la MPS 801. Elle possède l'avantage de se connecter au bus série IEEE du Commodore : pas de problème de compatibilité, sortie de tous les caractères graphiques. Par le jeu des adresses secondaires et des caractères de contrôle, on peut obtenir le contraste inversé, les caractères de double largeur, le code graphique, la tabulation, etc., ou encore définir des caractères programmables. Mais les minuscules n'ont pas de descendantes (matrice 7 x 5), elle est lente (30 caractères par seconde), elle est donc un peu limitée pour une utilisation courrier.

Si vous voulez connecter des imprimantes non Commodore, comme le bus IEEE série n'est pas standard, vous devrez recourir à une interface, qui peut vous ouvrir l'accès à tous les périphériques du monde, pour peu qu'ils soient compatibles.

Attention donc : avant de vous lancer dans l'achat d'un périphérique non Commodore, prenez un rendez-vous avec le vendeur, emportez votre C. 64 chez lui, et n'achetez que ce que vous aurez vu fonctionner.

Autres précautions : les interfaces nécessitent souvent un petit bout de programme pour les conduire. Sur certaines, ce programme est en mémoire morte et n'utilise pas le moindre octet en mémoire centrale. Sur d'autres, il faut le charger préalablement : on perd de la place en mémoire. Mais le prix s'en ressent. ■

François-Jean Bayard  
d'après Guy Ladevie et Daniel Trécourt



# SAUVEGARDE

## PROTÉGEZ VOS PROGRAMMES

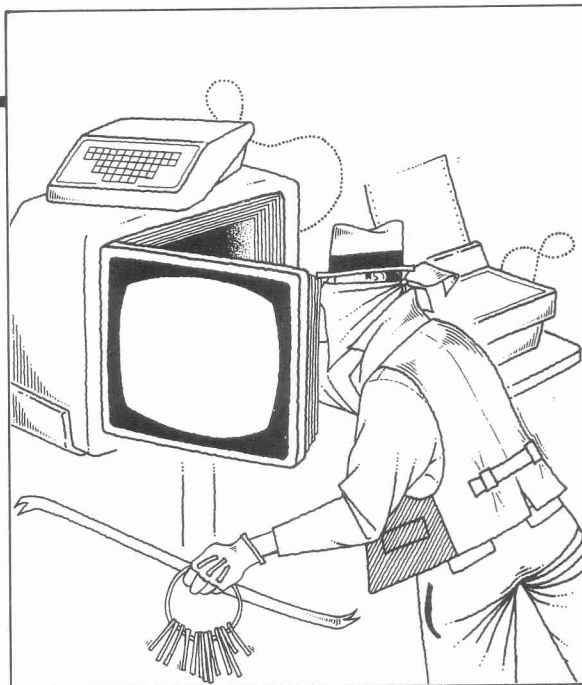
**Pirato-ci, Pirato-là, laissons tricorne et râpière au magasin des accessoires, et évoquons plutôt Arsène Lupin ou Harry Houdini. Car c'est bien de serrures qu'il s'agit, et de l'escalade historique du serrurier et du cambrioleur. Problème principal : se protéger des visiteurs et reproducteurs indéclicats.**

**S**i vous voulez éviter qu'un petit curieux ne s'approprie vos secrets de cuisine, ou simplement le résultat de vos longues veilles obstinées, sachez qu'il n'y a point de serrure inviolable. Tout dépend de la qualification du cambrioleur. Rappelons-nous que les auteurs ou diffuseurs qui veulent protéger leurs logiciels ont une double contrainte : la protection ne doit pas empêcher le produit vendu de tourner sur toutes les machines et le coût de la protection ne doit pas rendre le produit invendable.

En général, un bon moyen de protéger un programme sur C. 64 consiste à faire en sorte qu'il le lance dès son chargement et qu'on ne puisse en sortir qu'en éteignant l'ordinateur. Il suffit pour cela d'inhiber la touche STOP (POKE 808,239 ou POKE 788,52 ; rétablissement par POKE 801,1 et POKE 788,49), la combinaison STOP/RESTORE (POKE 808,225 ; rétablissement par POKE 808,1 ou plus méchant POKE 792,34 : POKE 793,253 essayez...), et chaque fois qu'on ne se sert pas du clavier de simuler un ON ERROR GOTO, en introduisant par POKE dans le tampon-clavier un GOTO numéro de ligne et un retour-chariot. Inconvénient : cela ne résiste pas à un bouton Reset, mais tout le monde ne s'en est pas équipé.

Une fois de plus, tout dépend du niveau présumé de votre cambrioleur ou plus exactement de son expérience. Car le bluff peut parfois suffire. Annoncez que votre programme sur cassette ne peut être lancé que par SHIFT/RUN, ce qui assure le lancement après le chargement. Votre curieux refuse-t-il d'obtempérer et se contente-t-il de LOAD ? Il perd curseur et clavier, conclut au « plantage », s'énerve, abandonne et c'est bien fait pour lui. Car le curseur est là. Mais invisible. Le truc ? En sauvegardant votre programme sur la cassette, au lieu de l'appeler « ZOZO », vous avez ajouté au nom, avant de refermer les guillemets, le caractère de couleur bleu. Lorsque votre client tape LOAD, l'ordinateur affiche FOUND ZOZO, mais il affiche aussi le dernier caractère, qui fera virer au bleu – couleur du fond de l'écran – le LOADING, le READY, et le curseur qui suit. Marrant, non ?

Pour imposer le lancement dès le chargement, la méthode la plus répandue consiste à utiliser lors de la sauvegarde l'adresse secondaire 1 (qui fait charger le programme à son adresse d'origine) et à inclure dans



le programme une partie de la page 1 qui comprend les vecteurs.

Une sécurité supplémentaire souvent mise en œuvre va vérifier, en cours d'exécution, la présence d'un certificat d'authenticité qui peut être un clef électronique en M.E.M. à brancher sur l'ordinateur, un caractère invisible (CHR\$(1) par exemple) dans le nom du programme, une inscription sur un secteur précis de la disquette, ou même une erreur de disque soigneusement programmée. Si ce témoin n'est pas là, un message menaçant ou goguenard prierà l'utilisateur de bien vouloir acheter ses programmes, avant de déclencher un Reset total de la machine, la sortie d'un gant de boxe au bout d'un bras articulé ou l'explosion d'une grenade offensive – tout à fait programmables en images !

Une bonne précaution consiste à dissimuler de son mieux la routine de vérification à l'intérieur du programme. S'il est en langage machine, c'est facile ; s'il est en Basic, l'utilisation judicieuse de caractères bizarres en REM (voir Trucastuces p. 63 de ce numéro) peut vous permettre de les cacher, mais le gag suprême vous est donné dans « Pas d'octet pour bernie soap » : il consiste à déguiser un programme Basic en programme LM. Si, c'est possible ! Vous n'avez qu'à aller y voir.

**François Chevalier**



# Le petit ordinateur illustré: votre magazine radio

15 minutes toutes les  
semaines, sur  
90 stations FM.



Ajaccio	R. Corse Int.	95.5 MHz	JEU	20h30
Albi	RTN	93.4 MHz	LUN	19h30
Alençon	AFM	89.4 MHz	MER	19h
Alès	Filasoi	88.6 MHz	JEU	19h
Amiens	RCC	101 MHz	SAM	11h
			MER	15h
Angers	Angers 101	101 MHz	MER	19h15
Angoulême	R. Marguerite	99.9 MHz	JEU	19h
Anncy	Sud Est Radio	102 MHz	SAM	9h30
Aurillac	Radio 15	89 MHz	MAR	19h45
			JEU	11h45
			JEU	20h30
Bastia	R. Corse Int.	91.4 MHz	JEU	20h30
Bayonne	R. Adour Navarre	90.7 MHz	JEU	18h
Belfort	Radio Soleil	88.1 MHz	SAM	12h05
Besançon	RVF	98.1 MHz	SAM	9h50
Bordeaux	Radio 100	94.3 MHz	DIM	10h15
Boulogne/mer	RBL	91.6 MHz	MER	12h
Bourges	Recto-Verso	98 MHz	SAM	12h05
Brest	Radiogram	103 MHz	SAM	14h
Briançon	RBV	94 MHz	VEN	18h
Brive	R. Brive Licorne	95 MHz	MAR	18h30
Caen	Caen FM	96.8 MHz	DIM	10h
Cahors	Chabrot	96.2 MHz	SAM	13h
Cannes	Fréquence Sud	97.7 MHz	SAM	19h
Carcassonne	Radio 11	94.1 MHz	MER	8h45
Castres	R. Tarn Sud	97.5 MHz	SAM	10h
Chalons	Radio Chalons	98 MHz	MER	19h15
Chambéry	Fréquence Horizon	100.2 MHz	JEU	19h30
Chartres	Radio Loisirs n 1	97.3 MHz	SAM	10h
Chateauroux	Radio Reflex	99.7 MHz	SAM	13h45
Cherbourg	Radio Rustine	87.8 MHz	VEN	19h
Clermont-Ferrand	Station MU	96.2 MHz	DIM	11h
Colmar	Radio 100	100.2 MHz	MAR	19h30

Creil	FDL	100.3 MHz	MER	19h15
Dax	ACQS 95	95.1 MHz	MER	19h
Dijon	Radio 2000	90.7 MHz	VEN	19h30
Evreux	REV	89.1 MHz	VEN	19h15
Fréjus	REM	97.8 MHz	SAM	13h30
Gap	RTM	90 MHz	MAR	13h
Grenoble	RVI	96 MHz	SAM	18h45
Guéret	REM	100.1 MHz	SAM	11h
Haute-Loire	RCL	100.8 MHz	LUN	18h45
La Rochelle	R. La Rochelle	92 MHz	LUN	18h45
Lannion	Pays de Trégor	91.6 MHz	SAM	19h30
Laval	Perrine	101.3 MHz	MAR	19h30
Le Havre	MAEVA	103.4 MHz	JEU	20h45
Le Mans	FM 104	104 MHz	JEU	18h
Lille	Contact	93.4 MHz	JEU	22h
Limoges	HPS	102.7 MHz	DIM	9h45
Lons-le-Saulnier	Lacuzon	89 MHz	MER	17h30
Lorient	AVB	102.7 MHz	MER	13h15
Lyon	Ciel FM	96.9 MHz	DIM	10h
Mâcon	Radio Pélican	88 MHz	MAR	12h30
Marseille	Fréq. Marseille	94.7 MHz	DIM	10h30
Metz	RADIO L	93.3 MHz	JEU	19h15
Mont-de-Marsan	Radio 6	97.9 MHz	SAM	9h45
Montélimar	R. Côtes du Rhône	104 MHz	MER	16h45
Montluçon	Caroline	96.2 MHz	SAM	12h
Montpellier	88.8	88.8 MHz	JEU	19h
Mulhouse	Bollwerk	88.1 MHz	MAR	21h30
Nancy	Rockin'Chair	95.8 MHz	SAM	12h
Nantes	Atlantic FM	96.8 MHz	MER	19h15
Narbonne	Radio Corail	93.6 MHz	SAM	9h30
Nîmes	Canal 30	101.5 MHz	TEL	64.31.87
Orléans	96.4	96.4 MHz	SAM	11h30
Paris	Gilda	103.5 MHz	DIM	10h
Pau	Fréquence 4	93.1 MHz	SAM	18h30
Périgueux	Fréquence 101	101.2 MHz	JEU	18h45
Perpignan	Stéréo 66	93.5 MHz	DIM	9h
Poitiers	RPO FORUM	90 MHz	MER	18h45
Reims	88.6	88.6 MHz	VEN	22h
Rennes	RBS	89.1 MHz	SAM	10h30
Rodez	Cité 12	101.6 MHz	MER	19h
Saint-Etienne	Transat	93.4 MHz	MER	19h
Saintes	Saintes 102	102 MHz	MER	19h
Saint-Gaudens	R. Comminges	92.9 MHz	MER	13h30
Saint-Lo	R. Manche	102.5 MHz	MER	18h30
Saint-Malo	Force 7	95.9 MHz	MAR	22h
St-Quentin	Azur 100	100.2 MHz	SAM	9h
Salon de Prov.	R. Centuries	99.7 MHz	LUN	18h30
Seine et Marne	RBS	103.7 MHz	SAM	13h
Sens	R. Horizons	91.2 MHz	MER	19h30
Strasbourg	Nuée Bleue	89.5 MHz	JEU	18h30
Tarbes	Pirène 98	98 MHz	SAM	8h45
Toulon	Mistral	104 MHz	SAM	10
Toulouse	Cambos	93.5 MHz	JEU	19h15
Tours	Méga Tours	103 MHz	JEU	12h45
Troyes	Discone Radio	92 MHz	JEU	18h45
Valence	Valence FM	91 MHz	MER	19h30
Vezoul	RVS	91.3 MHz	MAR	19h45
Vill. de Rouergue	Radio Villefranche	101.3 MHz	SAM	13h30

**Annonceurs, faites connaître votre produit  
ou vos services aux auditeurs de P.O.I.  
Contactez FORCE 7 au (1) 240.22.01.**

**poi** ordinateur



# ASSEMBLEUR

## LES OBJETS MANIPULABLES EN L.M.

**Contrairement au Basic, le L.M. (langage machine) court très vite. Cette différence provient du fait que le Basic prend son temps pour manipuler des variables ; l'interpréteur passe des heures entre les variables et les organes proprement dits de la machine. On manipule directement les éléments internes au C. 64. Ceux que nous allons manipuler sont de deux catégories : les cases mémoires et les registres.**

**R**appelons ce qu'est une case mémoire : en gros un emplacement capable de conserver une information. Nous employons à dessein le terme « emplacement », car une case mémoire est précisément définie par sa localisation géographique à l'intérieur de la machine. On dit qu'on donne son adresse. L'adresse est un nombre. Au point de vue logique, elle représente le numéro d'ordre de la case correspondante, comme si on considérait que toutes les cases étaient rangées à la suite les unes des autres. D'un point de vue physique, l'adresse permet de retrouver l'emplacement de la case, exactement comme une adresse postale permet de retrouver un immeuble.

Pour accéder à une case mémoire, il faut et il suffit d'indiquer son adresse. Quelles opérations peut-on faire sur une case mémoire ? Il n'y en a que deux : on peut lire le contenu de la case, ce qui revient à en prendre connaissance pour transmettre l'information à un autre organe de la machine. On peut écrire dans la case, ce qui revient à y ranger une information qui vient d'un autre organe.

Il faut noter que la lecture conserve l'information qui est dans la case mémoire : il y a une simple prise de connaissance, tandis que, bien sûr, l'écriture fait « oublier » l'information qui était présente dans la case avant l'opération.

Combien de temps ce travail prend-il ? Voilà une occasion de mesurer la vitesse du langage machine par rapport à celle du Basic. Une telle écriture met normalement 1 microseconde (1 millième de seconde). Comptons même 5 microsecondes pour pouvoir faire quelque chose avec la donnée lue. En Basic, l'équivalent de la lecture directe en mémoire qu'est la fonction PEEK a un temps d'exécution de 5 millisecondes, soit mille fois plus.

(N.B. : d'autres opérations donnent des gains moins spectaculaires.)

Eh bien, dans chaque case mémoire, on met un nombre exprimé en binaire sur 8 bits. Chaque bit ou chiffre binaire est capable de valoir 0 ou 1. Maintenant, de même que dans le système décimal (base 10) le nombre 1789 vaut  $9 + 8 \times 10 + 7 \times 10^2 + 1 \times 10^3 = 9 + 80 + 700 + 1000$ , en binaire (base 2), par exemple, le nombre 11011001 vaut  $1 + 0 \times 2 + 0 \times 2^2 + 1 \times 2^3 + 0 \times 2^4 + 0 \times 2^5 + 1 \times 2^6 + 1 \times 2^7 = 1 + 0 + 0 + 8 + 16 + 0 + 64 + 128 = 217$ .

Nous reviendrons sur ces codages binaires. Retenons que chaque case mémoire contenant 8 bits (c'est pour cela qu'on l'appelle aussi « octet ») la case mémoire peut renfermer un nombre de 0 à 255, ce qui correspond à 256 possibilités ( $2^8=256$ ).

Attention, la donnée peut avoir d'autres interprétations que l'interprétation numérique (ex : chaîne de caractères...). C'est là une des difficultés de l'utilisation du langage machine. En Basic ou dans tout autre langage évolué, on n'a pas à se préoccuper du codage des données.

Au cœur du microprocesseur se trouvent les registres, là où se traite l'information. Les registres sont des mémoires. Disons plutôt qu'ils remplissent, comme les cases mémoires, une fonction de mémorisation d'information ; mais ils



### Le revers de la médaille

Il nous faut voir maintenant le revers de la médaille. Quelle information met-on dans une case mémoire ?



diffèrent de la simple mémoire sur deux points :

- Ils sont d'accès plus rapide que la mémoire : l'unité centrale accède d'emblée aux données des registres puisqu'ils en font partie.

- On ne fait que lire et écrire le contenu des registres, ce qui correspond à des transferts purs et simples d'informations. On effectue aussi des opérations arithmétiques et logiques sur leur contenu, ce qui correspond à l'élaboration d'informations nouvelles à partir d'anciennes. De fait, les registres sont les seuls organes qui permettent les calculs.

Le principal registre arithmétique du 6502/6510 est l'accumulateur appelé A. Il fait 8 bits. La plupart des instructions de calcul sont de la forme « effectuer telle ou telle opération entre l'accumulateur et une case mémoire dont on vous indique l'adresse ». Le résultat se retrouve dans A. Dans le 6502, l'accumulateur est pratiquement le seul registre qui ait ces possibilités arithmétiques. Il existe aussi des instructions de transfert entre l'accumulateur et une case mémoire de la forme « amener dans A le contenu de la case d'adresse nnnn » et « ranger le contenu dans la case nnnn ».

Il en résulte que, pour faire une simple addition de deux nombres situés dans les cases mémoires d'adresses mmmm et nnnn, et si l'on veut que le résultat soit rangé à l'adresse rrrr, il faudra les trois instructions suivantes :

- « amener dans A le contenu de mmmm »
- « addition entre A et le contenu de nnnn »
- « ranger le contenu de A (c'est le résultat) à l'adresse rrrr ».

Plus compliquée à penser que l'instruction Basic  $R = M + N$ , cette opération est beaucoup plus rapide à l'exécution. Il faut d'ailleurs noter deux choses : • le 6502/6510 nécessite en fait une 4<sup>e</sup> instruction en raison de la façon particulière dont il fait les additions • l'addition que nous avons effectuée ici faisait intervenir des nombres de 8 bits. Pour être réellement équivalent à l'instruction Basic  $R = M + N$ , il aurait fallu un programme bien plus long.

### Instructions L.M.

Un programme, quel que soit son langage, réside en mémoire, dans des octets successifs. Chaque instruction, codée convenablement, va occuper un ou plusieurs octets (3 au maximum sur le 6502/6510). Le codage d'une instruction de la forme : « effectuer telle ou telle opération entre A et la case d'adresse mmmm » comportera deux parties : • le codage de la nature de l'opération à effectuer • le codage de l'adresse mémoire concernée.

Sur le 6502, le codage de l'opération (on l'appelle le code opération) tient toujours sur un seul octet de l'instruction et c'est le premier octet de l'instruction. L'adresse, elle, occupe zéro, un ou deux octets selon le mode d'adressage. Le cas zéro octet correspond à l'absence d'adresse, ce qui est le cas des instructions qui ne font pas intervenir la mémoire, mais les seuls registres internes. Les modes d'adressage font l'objet de plusieurs articles. Dans le cas le plus naturel, celui où l'on spécifie l'adresse purement et simplement, celle-ci occupe deux octets ou

16 bits afin de pouvoir couvrir un espace adressable de 64 K octets. S'il n'y avait qu'un octet d'adresse, on n'aurait que 256 octets possibles !

Le fait que le code opération soit le premier octet permet au microprocesseur de tout de suite savoir, lorsqu'il examine une instruction, combien d'octets il devra encore lire pour connaître l'instruction dans son ensemble.

Faisons maintenant le tour du propriétaire des registres internes du 6502/6510. Tous font huit bits sauf PC qui en a seize. Ils obéissent au schéma ci-dessous :

A	Registre d'état
Accumulateur	S
X	Pointeur de pile
Registre index	PC
Y	PCH
" "	PCL
P	Compteur ordinal

Ce schéma s'intitule le modèle de programmation, car on n'y a fait figurer que les registres « programmables », c'est-à-dire manipulables :

**A** est l'accumulateur, principal registre de calcul.

Nous avons déjà vu son rôle.

**X** et **Y** sont les registres index. Ils sont munis de quelques possibilités arithmétiques (incrément-décément). Leur rôle principal est de servir d'indices pour des tableaux ou de compteurs de boucles.

**S** est le pointeur de pile : son rôle sera vu à propos des sous-programmes et des interruptions.

**P** rassemble sept indicateurs d'état (le bit 5 est inutilisé). Chacun de ses bits est représentatif d'une situation qui a eu lieu dans la machine ou d'un mode de fonctionnement qui va influencer sur le comportement ultérieur suivant le schéma :

7 6 5 4 3 2 1 0  
n v b d i z c

**n**(état) « négatif » à 1 lorsque la dernière opération a donné un résultat négatif.

**v**(état) « débordement » à 1 lorsque la dernière opération a donné un dépassement de capacité.

**b**(état) « break » à 1 lorsqu'on vient d'effectuer une instruction BRK.

**d**(mode) « décimal » à 1 lorsqu'on est en mode décimal.

**i**(mode) « inhibition interruptions » à 1 lorsque les interruptions IRQ sont inhibées.

**z**(état) « zéro » à 1 lorsque la dernière opération a donné un résultat nul.

**c**(état) « retenue » (carry) à 1 lorsque la dernière opération a produit une retenue.

**PC** est le compteur ordinal (Program Counter).

C'est le seul registre seize bits. Il contient à tout moment l'adresse de la prochaine instruction à exécuter. Il est normal qu'il ait seize bits pour que les adresses puissent couvrir les 64 K d'espace adressable.

Voilà terminé ce petit tour des objets manipulables en langage machine. Pour aller plus loin, reportez-vous à vos ouvrages (voir liste page 65) ou aux numéros de *l'Ordinateur Individuel* (voir liste pages 63, 64) préférés, et amusez-vous avec Basic-liste, au cœur de ce C. 64 spécial. **D.-J. D.** ■



**2 MAGASINS :**  
ouverts du lundi  
au samedi

PLACE D'ITALIE  
62, rue Gérard - 75013 PARIS  
Tél. (1) 581.51.44  
(ouvert de 9 h à 19 h)

**RUN**  
INFORMATIQUE

RÉPUBLIQUE  
5, bd Voltaire - 75011 PARIS  
Tél. (1) 338.96.31  
(ouvert de 10 h à 19 h)

**LES "PRO" DU 64 LOGICIELS ET MATÉRIELS**

**LE COIN DES AFFAIRES**

**RAID OVER MOSCOW** K7 90,00 F  
**BEACH HEAD** K7 89,00 F

**LES DEUX** .. Réf. Afs 001 ~~179€~~ pour 169,00 F

**CONSULTEZ-NOUS**  
pour vos achats de 64...  
nombreuses  
PROMOTIONS

**DU NOUVEAU  
... TOUJOURS  
DU NOUVEAU...**

**A VOUS DE JOUER !!!**

**SPECTRUM**

**GREMLINS** Jeu d'aventure graphique inspiré du fameux film.  
64 Réf. 2457 (K7) ..... 157,00 F  
SPECTRUM Réf. 40098 (K7) ..... 157,00 F

**WORLD SERIES BASE BALL**  
Du sport US. On s'y croit!  
64 Réf. 2446 (K7) ..... 120,00 F  
SPECTRUM Réf. 40097 ..... 120,00 F

**GROG'S REVENGE**  
(la suite de BC QUEST, encore plus dingue!).  
Réf. 2458 (K7) ..... 130,00 F  
Réf. 2465 (Disk) ..... 169,00 F

**BOUNCY BOB STRIKES BACK**  
Réf. 2458 (K7) ..... 130,00 F  
Réf. 2465 (Disk) ..... 169,00 F

**DROPZONE, Fabuleux, du super galactique!**  
Réf. 2460 (K7) ..... 130,00 F  
Réf. 2461 (Disk) ..... 169,00 F

**DOUGH. Du super SYNISOFT.**  
Réf. 2462 (K7) ..... 130,00 F  
Réf. 2463 (Disk) ..... 169,00 F

**HUNCHBACK AT THE OLYMPICS**  
Réf. 2464 (K7) ..... 119,00 F

**SENTINEL**  
Réf. 2313 (K7) ..... 130,00 F  
Réf. 2314 (Disk) ..... 169,00 F

**SPITFIRE ACE**  
2337 (K7) 160,00 F 2338 (Disk) 190,00 F

**F 15 STRIKE EAGLE**  
2115 (K7) ..... 195,00 F

**JEU D'ÉCHECS**  
**COLOSSUS CHESS :**  
2073 (K7) 170,00 F 2074 (Disk) 220,00 F

**microlibrary**

**CBM 64**  
LES JEUX D'AVENTURE, COMMENT LES PROGRAMMER 65510 129 F  
PEEK'S ET POKES 65504 99 F  
LE LIVRE DE L'IMPRIMANTE 65503 179 F  
LE LANGAGE MACHINE DU COMMODORE 64 Tome 1 65505 149 F - Tome 2 65506 149 F  
TRUCS ET ASTUCES POUR LE 64 65507 149 F

**CBM 64 et VIC 20**  
Le livre du lecteur de cassette 1530, VIC-64 65501 99 F  
Le livre du lecteur de disquette 1541, VIC 65502 179 F

**AZIMUTH 3000**  
FINIS LES PROBLÈMES DE CHARGEMENT

- Le coffret comprend :
- Une cassette d'alignement de la tête AZIMUTH 3000.
- Un jeu d'essai BANDANA CITY.
- Un mode d'emploi en FRANÇAIS.
- Un tournevis spécial.
- Deux flèches indicatrices.

Chargez vos jeux et programmes en toute quiétude! Réf.: 2444 130F

**VOUS VOLEZ PROGRAMMER SÉRIEUX !!!**

**VICTREE**  
VICTREE vous sera d'un précieux secours. Cette cartouche d'aide à la programmation vous apporte beaucoup de commande qui font défaut au BASIC du 64. Par exemple PRINT USING, TRACE ou RENUMBER. VICTREE vous fournit beaucoup de commandes de DISK, d'ÉDITION et de MISE AU POINT de vos PROGRAMMES. VICTREE, un MUST pour programmer SÉRIEUX! Documentation en français.

**BAISSE DE PRIX** Réf.: 2387 cartouche 550F

**VOUS VOLEZ TRAITER DES TEXTES !!!**

**VIZAWRITE Français azerty**  
• Fusion avec informations d'autres fichiers (Superbase-BUSICALC).  
• Connexion directe d'imprimantes à interface type CENTRONICS.  
• Fourni avec manuel d'utilisation en français et autocollants pour les touches.  
• Formatage simple et visible à l'écran, jusqu'à 240 caractères par ligne et 34000 caractères en mémoire de travail.  
• Toutes fonctions classiques : fusion, marges temporaires, tabulateurs, déplacement et remplacement de texte, recherche, copie, en-têtes, numérotation automatique, correction, insertion, etc.

**NOUVEAU PRIX** Réf. 2389 Disk 950 F

**TEXTPRO**  
Traitement de texte simple (pas d'accent) mais efficace  
Fonctionne avec des imprimantes COMMODORE et, par exemple la MT 80 de Mannesmann (avec PRINT LINK) programme à menu simple 2360 K7 170 F - 2361 Disk 220 F

**VOUS VOLEZ DES AIDES A LA PROGRAMMATION !!!**

**ZOOM** Moniteur langage machine très puissant. Parmi ses fonctions : "trace" rapide, transfert, compare, load, save, verify, assemble et disassemble 2411 K7 195 F 2412 Disk 235 F

**GRAPHIX** Rend facile l'utilisation de la haute résolution sur votre 64. En langage basique équipé de fonctions de type AUTO, DELETE, FIND et d'un moniteur langage machine. 2143 K7 195 F 2144 Disk 235 F

**BACK UP** Utilitaire vous permettant de réaliser des copies exactes de vos disquettes avec un seul lecteur 2029 Disk 195 F

**MIKRO ASSEMBLEUR** Un assembleur trois passes qui rend le langage machine aussi simple que le basic! Le code source que vous écrivez est entré comme un programme basic équipé de fonctions de type AUTO, DELETE, FIND et d'un moniteur langage machine. MIKRO est un bien bel outil! 2287 K7 490 F 2288 Disk 540 F

**PRINT LINK 64** Câble pour relier le 64 à toute imprimante parallèle (par exemple Epson) + logiciel permettant d'écrire comme une imprimante Commodore 2289 Disk 540 F

**VOUS VOLEZ DES PROGRAMMES ÉDUCATIFS !!!**

**CONJUGAISON FRANÇAISE**  
Un logiciel conçu par une enseignante. Un logiciel actif pour un élève ou une classe. Méthode active pour un élève ou une classe. Le moniteur choisit le verbe que l'élève doit conjuguer. Le clavier et l'écran sont accouplés. 2081 K7 140 F - 2082 disquette 170 F

**TO SPEAK - I SPOKE**  
Les fameux verbes irréguliers anglais. De la 6<sup>e</sup> à la terminale. 3 niveaux : débutants, moyens, forts. Dictionnaire. Impression de la liste sur imprimante. 2005 K7 140 F - 2005 disquette 170 F

**MODEM DTL 2000**  
**MODEM DTL 2000+**

- DTL 2000 : carte V23 (1200/75 bauds) - interface et logiciel Minitel.
- DTL 2000+ : identique au précédent + carte V21 (300 bauds) permettant l'utilisation du réseau TRANSPAC.

En plus de la K7 de logiciels DIGITELEC, tous les MODEMS sont livrés avec NOS LOGICIELS complémentaires de communication et de hard copy imprimante (Disk). Vous tirerez ainsi le meilleur parti de votre DIGITELEC!  
DTL 2000 ..... Réf. 2234 1600 F  
DTL 2000+ ..... Réf. 2433 2100 F

**VOUS VOLEZ GÉRER !!!**

**BUSICALC** UN VRAI - CALC - PROFESSIONNEL. La combinaison de toutes ses caractéristiques et fonctions lui confèrent un rapport qualité/prix exceptionnel. Plus l'on s'en sert et plus l'on découvre de nouvelles applications. Il n'arrête pas de s'auto-améliorer.  
VIC 20 : 6001 K7 299 F - 6501 Disk 330 F  
C 64 : 2053 K7 299 F - 2054 Disk 339 F  
CBM 4032 6003 Disk 630 F  
CBM 8032 6004 Disk 630 F

**BUSICALC 2** Simple à utiliser, ultra-performant! Possesseur de 64, vous avez maintenant le choix. BUSICALC ou BUSICALC 2. Choisissez la nouvelle version. Utilisez totalement les possibilités mémoire du 64, ce qui permet de travailler d'autres fonctions, elle conserve la simplicité d'utilisation de la première. BUSICALC 2 ne possède pas de page HELP car il n'en a pas besoin.  
6601 K7 - 6002 Disk 600 F

**BUSICALC 3** Un des calculs les plus puissants du monde pour le 64. BUSICALC 3 possède toutes les fonctions de BUSICALC 2 + 3 dimensions + capacité de fonctionnement avec programmes « colonnes de largeur variable » historigrammes et EASYSCRIPT « un mot de passe pour protéger les fichiers » envoi des caractères spéciaux à l'imprimante, par exemple : caractères condensés. 2057 Disk 975 F

**SUPERBASE 64** Base de données pour le 64. Ce programme permet de gérer des ensembles de fichiers. Chaque fiche peut comprendre 1000 caractères qui sont présentés sur quatre écrans. Gestion de la base de données facile par menus. 2354 Disk 1180 F

**DATAPRO** La plus simple des bases de données et une des plus performantes pour son prix. Avec le même programme, vous pouvez l'utiliser pour mettre sur fichier vos collections de disques... vos programmes. Les commandes puissantes de DATAPRO vous permettent de rechercher rapidement toute "fiche" de la base de données. 2094 K7 170 F - 2095 Disk 220 F

**VOUS VOLEZ CONSERVER UNE IMAGE PAPIER DE VOTRE ÉCRAN !!!**

**PHOTOMATIC 64** (Hervé LE MARCHAND). Logiciel français de copie d'écran sur imprimante de programmes (dont ceux en haute résolution graphique) et compris les sorties. Deux formats d'impression (100 mm et 200 mm). Utilisable avec imprimantes Commodore, sauf MPS 8021 parallèles Centronics. Imprime les dessins réalisés avec PAINT MAGIC, DOODLE, KOLAPAD. 2122 Disk 195 F

**VOUS VOLEZ DES EXTENSIONS !!!**

**8002 16 K (cartouche SRC 16)** 549 F  
**8100 CARTE MÈRE** 349 F  
3 CARTEAUX

**(cartouche SRC 32) 8003 32 K**  
Le VIC 20 affiche \*\*\* 28159 BYTES FREE \*\*\* et donne au système 8192 octets utilisables en code machine. 849 F

**VOUS VOLEZ ÊTRE PLUS PERFORMANT !!!**

**joysticks - ATARI - CBM 64 - AMSTRAD - VIC 20**

**Le POWERPLAY de PROTEK.** Aussi rapide et plus solide que le "XXXXSHOT II". Un classique pour super Zapper!!!  
Réf. 2195 ... 150 F

**SLICKSTICK** Le joystick qui travaille tout en finesse. Une sensibilité extraordinaire.  
13010 205 F

**COMPETITION PRO** Le joystick "Super Classe"! Équipé de microswitches, il allie précision, douceur et fiabilité.  
13050 265 F

**ARCADE PROFESSIONAL**  
Le nec plus ultra dans l'arcade! Avec lui rien ne vous résistera et vous réaliserez des scores

**en promo**  
13055 450 F

**BON DE COMMANDE à découper et à renvoyer à CREDIT CETELEM**

LOGICIEL JEUX Qté \_\_\_ N° \_\_\_ Qté \_\_\_ N° \_\_\_  
LOGICIEL GESTION Qté \_\_\_ N° \_\_\_ Qté \_\_\_ N° \_\_\_  
BUSICALC Qté \_\_\_ N° \_\_\_  
EXTENSION Qté \_\_\_ N° \_\_\_  
MICROLIBRARY N° \_\_\_\_\_ Matériel \_\_\_\_\_

Je désire recevoir votre super catalogue "Commodore". 12,00 F Remboursement à la 1<sup>re</sup> commande.

Tous les prix comprennent la T.V.A. et les frais de port postaux (France Métropolitaine).

Signature des parents pour les moins de 18 ans \* France métropolitaine \* sauf pour les livres : 6 F de port par ouvrage.

Je préfère régler par Carte Bleue N° de Carte Bleue: \_\_\_\_\_

Expire à fin ---/---/---

Date de commande : \_\_\_\_\_  
Signature obligatoire: \_\_\_\_\_

Port GRATUIT\* Total \_\_\_\_\_

# MANETTES

## LUTINS POIGNEES ET COLLISIONS

**Tant dans les « arcades » que sur votre machine il y a des objets essentiels : les lutins. Nous vous présentons leur mise en œuvre sur un exemple simple et facile à réaliser.**

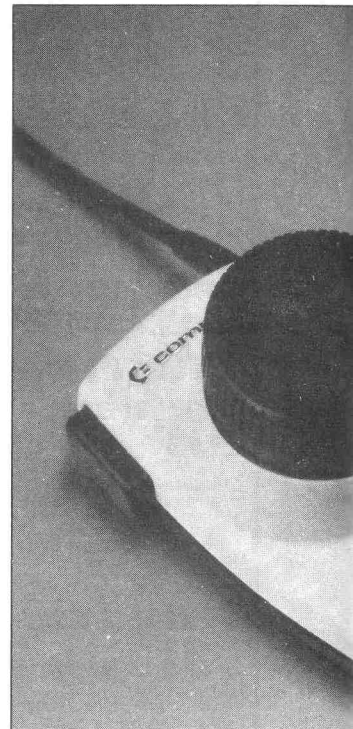
Les O.G.D. (Objets Graphiques Déplaçables pour les « restons français », lutins pour les « romantiques » ou sprites pour les « lecteurs anglophiles ») sont un des éléments essentiels de l'animation sur l'écran. L'O.G.D., appelons-le lutin, est une forme graphique définie en haute résolution dans une matrice 21 lignes sur 24 colonnes. Celui du C.64 reste un précurseur casse-tête intéressant.

Pour lui donner vie, la première manœuvre consiste à en définir la forme. Pour chaque sprite, il faut 63 octets disposés comme ci-dessous :

octet 0	octet 1	octet 2
3	4	5
6	7	.
.	.	.
.	.	.
.	.	.
60	61	62

Pour savoir quelles données mettre dans les octets, vous devez dessiner la forme que vous souhaitez sur une grille 21 x 24, ou plutôt trois grilles 21 x 8 juxtaposées. Chaque point allumé

Des poignées aussi vieilles et efficaces que le Commodore 64.



donne un bit 1 dans un octet. A titre d'exemple, faisons-nous un sprite en forme de ballon en suivant la figure 1.

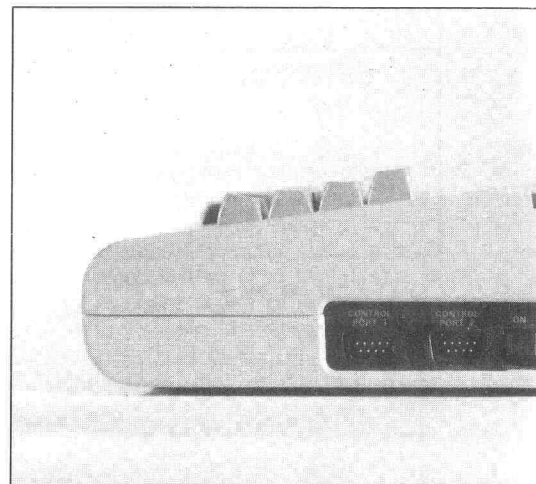
**Choisir une zone inaccessible à Basic puis numéroté les lutins**

Il ne reste plus, cette opération effectuée, qu'à placer ces données judicieusement en mémoire. Cela se fait en deux temps. On décide d'abord de la zone d'adresses. Pour ce premier exemple, nous choisissons la zone qui commence en 12288 (3000 en hexadécimal). Il faut protéger cette zone en la rendant inaccessible à Basic. Pour cela, la première instruction effective de votre programme doit être :

Figure 1	binaire			et les symétriques		
	0	0	0			
	0	0	0			
	0	0	0			
	0	01111110	0			
0000001	11111111	10000000		0	62	0
0000011	11111111	11000000		1	255	128
0000011	11111111	11000000		3	255	192
0000111	11111111	11100000		3	255	192
0000111	11111111	11100000		7	255	224
0000111	11111111	11100000		7	255	224
0001111	11111111	11110000		15	255	240
0001111	11111111	11110000		15	255	240

D'où les DATA correspondantes :

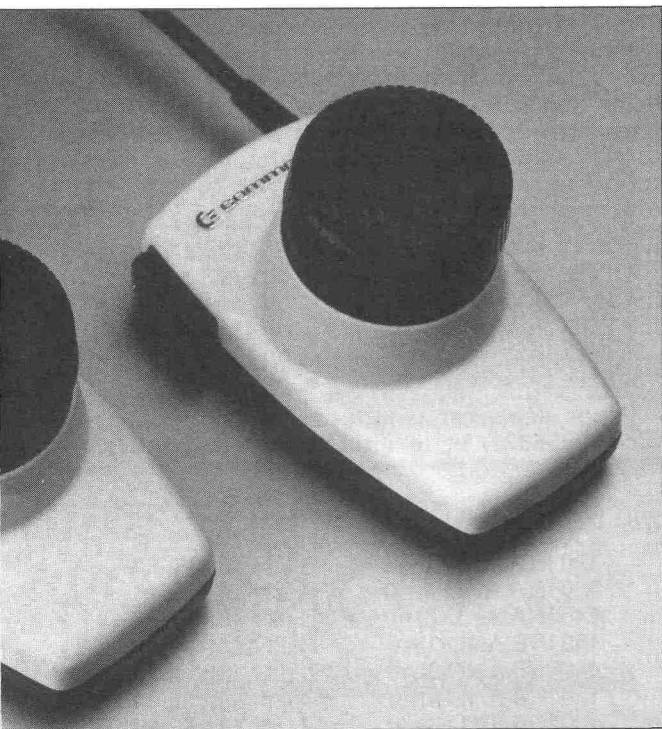
1000 DATA	0, 0, 0, 0, 0, 0, 0, 0, 0
1010 DATA	0, 62, 0, 1, 255, 198, 3, 255, 192, 3, 255, 192
1020 DATA	7, 255, 224, 7, 255, 224, 15, 255, 240, 15, 255, 240
1030 DATA	15, 255, 240, 15, 255, 240, 7, 255, 224, 7, 255, 224
1040 DATA	3, 255, 192, 3, 255, 192, 1, 255, 128, 0, 620, 0, 0, 0, 0, 0



Deux portes sur le côté droit, mais un seul manche à balai suffit pour jouer.

16





10 POKE 52, 48 : POKE 56, 48 : CLR  
(48 soit 30 en hexadécimal représente la partie haute de l'adresse).

Ensuite, il faut numéroté les lutins. Soit I le numéro. I peut aller de zéro à sept : il y a huit lutins possibles. Cela étant, il faut placer en 2040 + I un pointeur qui fixe l'adresse des octets de définition du lutin n° I.

Ici, nous ferons pour le lutin numéro zéro :

20 POKE 2040, 192

192 est une valeur magique qui résulte de l'adresse \$3000 par une cuisine que nous n'expliquerons pas ici.

Supposons que nous voulions un 2<sup>e</sup> ballon sur l'écran.

Nous ferons :

30 POKE 2041, 192

2041 = 2040 + 1 car c'est le lutin n° 1. 192 car on utilise le même dessin que le lutin 0. La définition de nos deux lutins se fera alors par :

40 M = 12288 : FOR J = M TO M + 62

50 READ K : POKE J, K : NEXT

Il ne reste plus qu'à définir d'autres éléments pour nos lutins, ce qui se fait par des POKE dans les registres du boîtier de gestion d'écran du 64, le 6567. Les adresses sont comprises entre 53248 et 53294 (\*).

Le lutin n° I est actif si le bit I du registre d'adresses 53269 est à 1. Il est invisible si ce bit est à 0. Donc pour faire apparaître le lutin n° I, on tape :

POKE 53269, PEEK (53269) OR 2 ↑ I

Cette instruction n'affecte que le bit I donc ne trouble pas les autres lutins éventuels.

Pour définir la couleur du lutin n° I, on frappe :

POKE 53287 + I, code couleur

Les codes couleurs de 0 (noir) à 15 (gris clair) sont les codes habituels.

Pour les couleurs obtenues à l'aide de la touche CTRL, le code est le chiffre de la touche -1. Pour les couleurs obtenues à l'aide de la touche C=, le code est le chiffre de la touche +7.

Exemple : noir : CTRL 1 donc code 0

bleu clair : C= 7 donc code 14

Ainsi, si nous voulons activer nos deux lutins et avoir le lutin 0 rouge et lutin 1 vert clair nous aurons :

60 POKE 53269,3

3 en binaire = 00000011

└ bit 0 = lutin 0 actif

└ bit 1 = lutin 1 actif

70 POKE 53287,2 : POKE 53288,13

Si vous entrez les instructions numérotées et faites RUN, vous espérez voir deux ballons à l'écran. En fait, vous risquez fort de ne rien voir du tout si vous n'avez pas fixé la position de vos sprites sur l'écran.

### Le positionnement en coordonnées X et Y

Le positionnement s'effectue en donnant les coordonnées X et Y du bord gauche du lutin sur l'écran suivant le schéma ci-après. Notons qu'il y a une marge de visibilité. On définit X de 0 à 347 et Y de 0 à 250. Pour que le lutin soit visible sur l'écran (donc à l'intérieur du cadre), il faut que X soit compris entre 24 et 321 et Y entre 50 et 229.

0	347	X
250	Y	Zone invisible

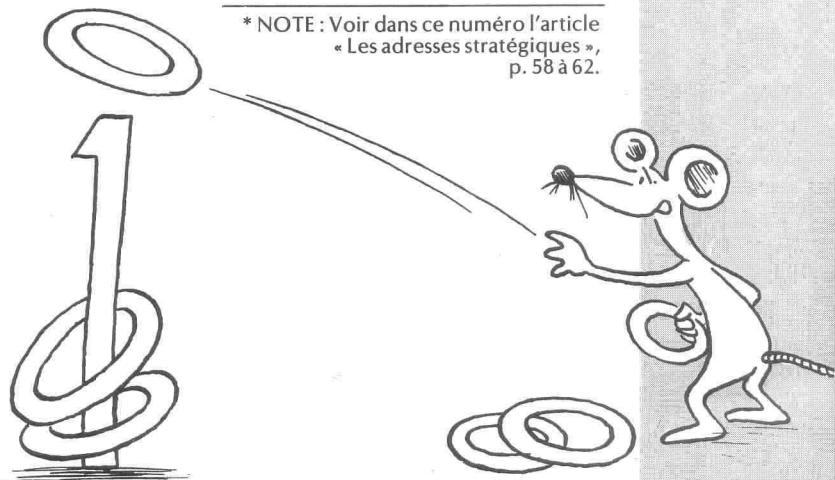
Rien de plus facile que de fixer Y, puisque  $Y < 255$  : il suffit d'un POKE 53249 +  $2 \times Y$ .

Pour X, c'est plus difficile car le nombre tient en 9 bits. Les 8 bits les plus bas forment la valeur  $X \text{ AND } 255$ . Cela est à mettre par POKE en 53248 +  $2 \times I$  (I = numéro de lutin). Le bit le plus élevé peut s'exprimer par  $-(X > 255)$  puisque la parenthèse vaut -1 si la condition est vraie, et 0 dans le cas contraire. Cette valeur doit être imposée comme lème bit du registre 53264. Cela se fait par :

POKE 53264, (PEEK (53264) AND (255 -  $2 \uparrow I$ ) OR ( $-2 \uparrow I$ ) × (X > 255))

Pour simplifier, nous resterons toujours dans les trois quarts à gauche de l'écran, ce qui sera

\* NOTE : Voir dans ce numéro l'article « Les adresses stratégiques », p. 58 à 62.



assuré en ne s'occupant pas du neuvième bit.

Assurons la visibilité de nos deux ballons par :  
80 X0 = 100 : Y0 = 120 : X1 = 200 : Y1 = 80  
90 POKE 53248, X0 AND 255 : POKE 53249, Y0  
AND 255 : POKE 53250, X1 AND 255 : POKE  
53251, Y1 AND 255

(cette ligne doit être entrée en abrégé pour pouvoir tenir). Si vous faites RUN, vos lutins apparaîtront. Si vous avez gardé la liste de votre programme sur l'écran (ou tout autre affichage), vous constaterez deux choses : les lutins s'affichent indépendamment de ce qu'il y a sur l'écran ; ils cachent les caractères qui sont dessous. On dit qu'ils ont priorité sur l' « arrière-plan ».

Maintenant, modifiez 80 pour avoir X1 = 105 et Y1 = 115. Les deux lutins se recouvrent en partie, et le rouge (lutin 0) cache le vert (lutin 1). Un lutin de numéro plus petit a priorité sur les lutins de numéro plus grand.

Comment déplacer vos lutins rapidement ? Grâce aux « joysticks ». Nous vous conseillons de vous acheter au moins un manche à balai : ce n'est pas très cher. Ici, nous en utilisons deux

```
10 POKE 52,48:POKE56,48:CLR
15 S0=0:S1=0:PRINT "S" S0,S1
20 POKE 2040,192
30 POKE 2041,192
35 FOR L=1 TO 20:FOR C=1 TO 30:A$=" "
36 IF RND(1)>.9 THEN A$="&"
37 PRINT A$;:NEXT:PRINT:NEXT
40 M=12288:FOR J=M TO M+62
50 READ K:POKE J,K:NEXT
60 POKE 53269,3
70 POKE 53287,2:POKE 53288,13
80 X0=100:Y0=120:X1=200:Y1=80
90 POKE53248,X0AND255:POKE53249,Y0AND255
:POKE53250,X1AND255:POKE53251,Y1AND255
100 A=PEEK(56320):B=PEEK(56321)
110 IF (A AND 1)=0 THEN Y0=Y0-10
120 IF (A AND 2)=0 THEN Y0=Y0+10
130 IF (A AND 4)=0 THEN X0=X0-10
140 IF (A AND 8)=0 THEN X0=X0+10
150 IF (B AND 1)=0 THEN Y1=Y1-5
160 IF (B AND 2)=0 THEN Y1=Y1+5
170 IF (B AND 4)=0 THEN X1=X1-5
180 IF (B AND 8)=0 THEN X1=X1+5
190 IF PEEK(53278) THEN S0=S0+1
195 IF PEEK(53279) AND 1 THEN S1=S1+1
200 PRINT"S" S0,S1
210 GOTO 90
1000 DATA 0,0,0,0,0,0,0,0,0,0
1010 DATA 0,62,0,1,255,128,3,255,192,3,2
55,192
1020 DATA 7,255,224,7,255,224,15,255,240
,15,255,240
1030 DATA 15,255,240,15,255,240,7,255,22
4,7,255,224
1040 DATA 3,255,192,3,255,192,1,255,128,
0,62,0,0,0,0,0,0,0,0
```

READY.

(c'est le maximum : il y a deux connecteurs). Même si vous n'en possédez pas encore, la suite de l'article vous intéressera. Les deux manches à balai que l'on peut connecter s'appellent 1 et 2 (numéros de leur connecteur). Le manche à balai définit quatre directions nord, ouest, sud et est et possède un bouton « feu ». On décode l'état du manche par PEEK à l'adresse 56321 pour le manche 1 et à 56320 pour le manche 2.

Si PEEK (adresse) AND 1 = 0, on a Nord Y=Y- 1  
Si PEEK (adresse) AND 2 = 0, on a Sud Y=Y+ 1  
Si PEEK (adresse) AND 4 = 0, on a Ouest X=X- 1  
Si PEEK (adresse) AND 8 = 0, on a Est X=X+ 1  
Si PEEK (adresse) AND -16 = 0, on a Feu

Nous déplacerons le lutin 1 à l'aide du joystick 1 (56321) et le lutin 0 avec le joystick 2 (56320). Ajoutons les instructions :

```
100 A=PEEK(56320) : B=PEEK(56321)
110 IF(A AND 1)=0 THEN Y0=Y0- 1
120 IF(A AND 2)=2 THEN Y0=Y0+ 1
130 IF(A AND 4)=0 THEN X0=X0- 1
140 IF(A AND 8)=0 THEN X0=X0+ 1
150 IF(B AND 1)=0 THEN Y1=Y1- 1
160 IF(B AND 2)=0 THEN Y1=Y1+ 1
170 IF(B AND 4)=0 THEN X1=X1- 1
180 IF(B AND 8)=0 THEN X1=X1+ 1
190 GOTO 90
```

Faites RUN. Vous voyez maintenant comment déplacer avec chacun des manches à balai le lutin qui lui correspond. Avec notre gestion simplifiée de X vous verrez que si vous déplacez un lutin à fond à droite, il disparaît puis réapparaît par la gauche.

Les déplacements sont précis. La lenteur est due à Basic qui passe un temps fou dans la série des tests 110 - 180. Peut-on accélérer ? Oui. On peut remplacer les +1 ou -1 de 110 à 180 par des +2, -2 ou même +5, -5. Cinq semble assez convenable. Au-delà, les déplacements seraient vraiment trop saccadés.

### Collisions en séries entre lutins

Lorsque deux lutins se télescopent, les deux bits du registre 53278 de numéros égaux aux numéros de ces lutins passent à 1. Il suffit de faire : IF PEEK(53278) AND (2↑I) THEN action, pour agir si le lutin n° 1 a fait une collision avec un autre lutin. Ajoutez les instructions :

```
15 S0 = 0 : S1 = 0 : PRINT "Clr" S0,S1
190 IF PEEK(53278) THEN S0=S0+1
200 PRINT "Home" S0,S1
210 GOTO 90
```

Vous voyez que S0 est augmenté à chaque contact des lutins. Comment augmenter S1 ? Facile. Mettons d'abord des « rochers » sur l'écran en y imprimant des caractères damiers.

```
35 FOR L=1 TO 20 : FOR K=1 TO 30 : A$ =
"espace"
36 IF RND(1)>.8 THEN A$="damier"
37 PRINT A$;:NEXT : PRINT : NEXT
```

Le 0.8 est à doser pour décider de la densité de rochers que vous voulez.

Maintenant, si le lutin I heurte un rocher, le



bit 1 du registre 53279 passe à 1. Ajoutez l'instruction :

```
195 IF PEEK (53279) AND 1 THEN S1=S1+1
```

Vous voyez que S1 augmente à chaque fois que le lutin rouge heurte un rocher.

### Un jeu de manche pour voir rouge

Décidons d'une règle du jeu. Chaque joueur commande un lutin par l'intermédiaire du manche à balai correspondant. Le sprite rouge poursuit le vert. A chaque fois qu'il rattrape le vert, le rouge gagne un point. En revanche, alors que le vert a le droit de passer sur les rochers, le rouge

donne un point à son adversaire à chaque fois qu'il heurte un rocher.

Passionnant non, ce jeu qui met aux prises deux adversaires ? Vous pourrez même le modifier pour jouer contre l'ordinateur, à condition de lui indiquer le mode de contrôle du lutin poursuivi.

Et le programme direz-vous ? Il est déjà écrit. C'est celui que vous venez de construire. Un seul problème, il reste à équilibrer les vitesses : le poursuivant doit courir plus vite que le poursuivi. Nous mettons donc des 10 en 110 à 140. Il faut aussi raréfier les rochers en mettant 0.9 ou même 0.95 en 36. Nous vous laissons faire ces dosages avec l'aide du listing. Bon jeu !

**Daniel-Jean David**

## DES TRUCS DE REFERENCE O.I.

### A VOS CASSETTES

J.-P. Lalevée

L'adresse 1 de la mémoire du Commodore 64 est un registre dont le bit n° 4 est à 1 lorsque le moteur du magnétophone ne fonctionne pas (toutes touches relâchées) et à 0 dans le cas contraire.

PEEK (1) AND 16 donne 0 si une touche de déroulement est appuyée sur le magnétophone.

PEEK (1) AND 16 donne 16 si toutes les touches sont relevées.

Vous pouvez utiliser cette particularité dans vos programmes interactifs qui demandent de la part de l'utilisateur une manipulation du magnétophone en cours de programmation. ■

### POUR UTILISER LES POIGNEES DE JEU

Franck Bonbled

Le Commodore 64 possède deux connecteurs à contacts utilisables avec des poignées de jeu ou un crayon lumineux.

Chacun d'eux accepte une poignée de jeu. Le crayon lumineux ne peut s'utiliser qu'avec le connecteur numéro 1.

Une poignée peut se définir comme cinq interrupteurs : un pour le bouton de tir et un pour chaque direction.

Les cinq interrupteurs correspondent aux cinq premiers bits des mémoires 56320 (port 2) et 56321 (port 1)

Exemple : Tir (interrupteur 4)  
En haut (0)

A gauche (2) — — A droite (3)

En bas (1)



Normalement, les bits sont mis à 1 quand une direction n'est pas choisie ou quand le bouton de tir n'est pas pressé.

Il suffit donc de tester les mémoires 56320 ou 56321 pour savoir les actions désirées par la poignée.

Le petit programme ci-dessous permet de

faire déplacer un lutin sur tout l'écran, à l'aide de la poignée connectée au port 1.

### PROGRAMME

```
7 REM - FRIT DEPLACER UN SPRITE SUR TOUT L'ECRAN
A L'AIDE
8 REM - D'UN JOYSTICK PLACER DANS LE PORT 1
10 PRINT «  »
20 AB=53248:WR=56321
30 POKE AB+32,8:POKE AB+33,5
40 POKE AB+21,1:POKE AB+39,6:POKE 2040,13
45 FOR G=832 TO 891:READ A:POKE G,A:NEXT
50 X=161:Y=128
60 M=PEEK(WA)
70 PRINT «  »;M
80 E=0:D=1
90 IF (M AND 16) = 0 THEN E=16:D=5
100 IF M=255-E THEN 60
101 IF M=254-E THEN Y=Y-D:GOTO 110
102 IF M=247-E THEN X=X+D:GOTO 110
103 IF M=253-E THEN Y=Y+D:GOTO 110
104 IF M=251-E THEN X=X-D:GOTO 110
105 IF M=246-E THEN X=X+D:Y=Y-D:GOTO 110
106 IF M=245-E THEN X=X+D:Y=Y+D:GOTO 110
107 IF M=249-E THEN X=X-D:Y=Y+D:GOTO 110
108 IF M=250-E THEN X=X-D:Y=Y-D
110 IF Y<29 THEN Y=250
115 IF Y>249 THEN Y=28
120 IF X<0 AND T=0 THEN X=90:T=1:POKE AB+16,1
125 IF X<0 AND T=1 THEN X=255:T=0:POKE AB+16,0
130 IF X>255 AND T=0 THEN X=0:T=1:POKE AB+16,1
135 IF X>91 AND T=1 THEN X=0:T=0:POKE AB+16,0
140 POKE AB,X:POKE AB+1,Y
150 GOTO 60
200 DATA 7, 255, 224, 15, 255, 240, 31, 255, 248, 63, 255, 252,
127, 255, 254, 252, 60, 63, 252
210 DATA 60, 63, 252, 255, 63, 252, 255, 63, 255, 255, 255,
255, 255, 255, 0, 255, 255, 0
220 DATA 255, 126, 126
230 DATA 126, 61, 255, 188, 29, 255, 184, 31, 255, 248, 31,
255, 248, 27, 187, 184, 17, 17, 16.
```

### Correspondances

Bouton de tir :  
bit 4=1 → pas de tir  
bit 4=0 → tir  
Direction en bas :  
bit 1=1 → non  
bit 1=0 → oui

Direction en haut :  
bit 0=1 → non  
bit 0=0 → oui  
Direction à gauche :  
bit 2=1 → non  
bit 2=0 → oui  
Direction à droite :  
bit 3=1 → non  
bit 3=0 → oui ■

**VOTRE ADRESSE ?**

Hervé Lemarchand

Connaître l'adresse de début d'un programme enregistré sur disquette ? Facile...

```
10 OPEN 1, 8, 2, « titre du programme »
20 GET #1, A$, B$
30 PRINT ASC (A$ + CHR$ (0)) + 256 × ASC (B$ + CHR$ (0))
40 CLOSE 1
```

Cela fonctionne, bien sûr, avec le Vic 20 ! ■

**ASTUCES PERIPHERIQUES**

Hervé Lemarchand

Faire une « copie » d'écran sur votre imprimante ? Facile... jugez-en :

```
10 OPEN 1,3 : OPEN 2,4
20 PRINT CHR$ (19); :FOR I = 0 TO 999 : GET #1, X$ : PRINT #2,X$; :NEXT I
30 CLOSE 1 : CLOSE 2
```

C'est parfait pour une imprimante 40 colonnes. Si la vôtre imprime sur 80 colonnes ou plus, la correction est simple :

```
10 OPEN 1,3 : OPEN 2,4
20 PRINT CHR$ (19); :FOR I = 0 TO 24 :FOR J = 0 TO 39 : GET #1, X$ : PRINT #2,X$; :NEXT J : PRINT #2 : NEXT I
40 CLOSE 1 : CLOSE 2
```

Pour le Vic 20, modifiez les valeurs de I et J, qui représentent respectivement le nombre de lignes et de colonnes de l'écran. ■

**DEUX PETITS SYS ET PUIS S'EN VONT**

Olivier Carré

Le *scrolling* vertical de l'écran (défilement des lignes vers le haut) est facile à obtenir par un simple SYS 59626. L'appel de cette routine déplace l'écran d'une ligne seulement. L'insertion de cette instruction dans une boucle vous permettra de « scroller » à loisir...

**Effacement de ligne**

Légalement plus complexe à réaliser, voici la clé à employer :

```
POKE 781, Y: SYS 59903
```

Formule dans laquelle Y représente le numéro de la ligne à effacer. (La première ligne porte le numéro 0.)

Fort de ces connaissances, testez votre opinion avec ce programme. ▼

**PROGRAMME**

```
100 PRINT « ☐ »: POKE 53280,0:POKE 53281,0
110 FOR T=1 TO 45:READ X
120 PRINT CHR$ (X);:NEXT T
130 DATA 18, 76, 39, 79, 82, 68, 73, 78, 65, 84, 69, 85, 82, 32, 73, 78, 68, 73, 86, 73, 68, 85, 69
140 DATA 76, 146, 32, 69, 83, 84, 13, 85, 78, 32, 66, 79, 78, 13, 77, 65, 71, 65, 90, 73, 78, 69
150 FOR T=0 TO 1000:NEXT T:PRINT « ☐ »
160 POKE 781,2:SYS 59903
170 FOR T=1 TO 11:READ X
180 PRINT CHR$ (X);:NEXT T
190 DATA 76, 69, 32, 77, 69, 73, 76, 76, 69, 85, 82
200 PRINT:PRINT « ERREUR CORRIGEE »
210 FOR T=0 TO 1000:NEXT T
220 FOR T=1 TO 10:SYS 59626:NEXT T
230 SYS 58260
240 END
READY.
```

**COLORIS DE PRINTEMPS**

Eric Draillard

POKE 646, A modifie la couleur des caractères affichés à l'écran. Le nombre A représente le code de la couleur, à savoir 0 = noir ; 1 = blanc ; 2 = rouge... De quoi colorier vos programmes. ■

**ERREUR DE LECTURE**

J.-P. Lalevée

Il arrive parfois qu'une tentative de chargement d'un programme à l'aide d'un magnétophone se solde par un rébarbatif LOAD ERROR, quelle que soit l'unité centrale utilisée (Vic 20, CBM 64 ou autre...) et, cela, malgré la réputation de fiabilité des enregistrements Commodore.

Le truc que nous vous proposons vous permettra de venir à bout facilement d'un type particulier – et fréquent – d'erreur de chargement.

Si donc, lors d'une mise en mémoire de programme, le fatidique message d'erreur apparaît, la première chose à faire est de vérifier la présence du programme en mémoire. Un simple LIST, tapé immédiatement après le LOAD ERROR, vous permettra de faire cette vérification.

Ensuite, ne tentez surtout pas de lancer le programme, sinon il serait immédiatement détruit ! En effet, la détection de l'erreur de chargement a interrompu brutalement le processus normal de mise à jour de quelques pointeurs. Votre machine ignore donc la présence d'un programme dans ses entrailles électroniques... Aussi, pour remédier à cette ignorance, un petit travail vous incombe.

Tapez, en mode direct, après avoir vérifié la présence complète du programme en mémoire, les six instructions suivantes :

```
POKE 45, PEEK(831)
POKE 46, PEEK(832)
POKE 47, PEEK(831)
POKE 48, PEEK(832)
POKE 49, PEEK(831)
POKE 50, PEEK(832)
```

Attention, prenez soin de ne commettre aucune erreur d'adresse dans ces six instructions !

Ensuite, vous pouvez vérifier (ce n'est pas indispensable), par la commande PRINT FRE(0), que le programme est alors pris en compte par la machine. Vous pouvez à présent lancer le programme normalement.

Si vous possédez un Commodore 3000, 4000 ou 8000, le même truc est utilisable, à condition de modifier les adresses en cause.

Remplacez les adresses 45 à 50 par les valeurs 42 à 47, et les adresses 831 et 832 par 637 et 638. Dans le cas où vous utilisez le port cassette n° 2, ces dernières valeurs sont 829 et 830.

Les valeurs données plus haut sont valables aussi bien pour le Vic 20 que pour le CBM 64.

Si la méthode proposée ne fonctionne pas, assurez-vous que le programme est bien entièrement en mémoire : ce qui est évidemment indispensable.

Dans l'affirmative, vous pouvez recommencer le processus décrit précédemment, mais en ne chargeant que l'en-tête du programme (en appuyant sur STOP dès que le message FOUND apparaît). ■



# GRAPHIQUES

## BOITE A OUTILS

**Voici un programme qui simplifie grandement les tracés Haute Résolution à partir du Basic du Commodore 64, qui – en version de base – souffre d'une carence chronique en matière d'instructions graphiques.**

**A**près mise en mémoire et exécution, le programme place le début du Basic 16384 (\$4000). Les quatorze Koctets de mémoire ainsi réservés seront utilisés en majeure partie par l'écran graphique (de 8192 à 16191 décimal). 2 300 octets seront occupés par la « boîte à outils » écrite en langage machine ; le reste demeure disponible pour des extensions personnelles. L'usage de cette « boîte à outils » reste transparent à l'utilisateur qui peut charger, sauvegarder et exécuter ses programmes de façon habituelle malgré la réduction normale de l'espace mémoire.

Le programme de base est proposé sous la forme d'un chargeur Basic contenant un nombre de DATA assez important. Comme toujours en pareil cas, on portera une attention particulière à leur transcription, pour éviter qu'un « plantage » intempestif ne vienne en perturber le fonctionnement. La prudence veut aussi qu'une sauvegarde soit effectuée sur cassette ou disquette avant lancement. Préalablement à la mise en mémoire chargeur, on déplace le pointeur de mémoire Basic sur l'adresse 16385 en tapant en mode direct au clavier :

POKE 43,1:POKE 44,64:POKE 16384,0:NEW

La frappe du programme peut alors commencer. Avant le RUN fatidique, on sauvegardera le programme Basic selon la méthode habituelle. RUN se fait ensuite sans risque pour placer les valeurs requises aux adresses prévues. Le résultat est un programme L.M (comportant une ligne Basic de lancement) qu'il faut sauvegarder à son tour en tapant en mode direct :

POKE 43,1:POKE 44,8:POKE 45,2:POKE 46,17  
puis : SAVE « ROUTINES H.R. » (,8 pour disquette). A partir de ce moment, c'est ce programme que l'on utilise sans oublier le RUN aussitôt après le chargement.

### Pas d'exécution graphique sans mémoire des coordonnées

L'exécution d'ordres graphiques est soumise au placement préalable en mémoire des coordonnées de référence avant appel des routines. La valeur maximale étant de 255, la dimension de l'écran graphique se réduit à 255x200 points.

–SYS 2096: commute le mode graphique.

–SYS 2122: efface l'écran graphique.

–SYS 2148: colore en blanc le fond de l'écran.

Pour changer de couleur, frapper POKE 2159, couleur : SYS 2148

–SYS 4315: sortie de mode graphique et retour à l'écran normal.

–POKE 8000,X:POKE 8002,Y:SYS 2180: allume le point de coordonnées X et Y.

–POKE 8000,X1:POKE 8002,Y1:POKE 8001,X2:POKE 8003,Y2:SYS 2340: trace une ligne reliant les points de coordonnées (X1,Y1) et (X2,Y2). Si X1,Y1 sont omis, le dernier point tracé est pris pour origine.

–POKE 8010,X:POKE 8011,Y:POKE 8012,RX:POKE 8013,RY:SYS 2513: dessine un cercle ou une ellipse de centre X,Y. RX et RY donnent la valeur du rayon selon les axes X et Y (ce qui permet de tracer les ellipses).

–POKE 8010,X:POKE 8011,Y:POKE 8012,RX:POKE 8013,RY:POKE 8015, début:POKE 8017,fin:SYS 2735: trace un arc de cercle seulement entre les valeurs de début et de fin, qui sont des valeurs angulaires données en grades (de 0 à 200). « Début » doit toujours être inférieur à « fin ». Si l'arc doit chevaucher le point 0, il faut le tracer en deux fois. Exemple : tracé d'arc de 175 à 25: début=175, fin=200, puis début=0, fin=25.

–POKE 8000,X:POKE 8002,Y:SYS 4200: efface un point situé en X,Y.

–SYS 4280: efface des cercles, des lignes, et s'utilise avec les mêmes POKE que ceux qui servent à les tracer.

–SYS 4297: revient au mode impression après l'usage du SYS 4280.

Pour se convaincre de l'efficacité de ce programme, on tapera le programme de démonstration qui dessine une carte de France sur l'écran. Pour conclure, voici un tableau contenant les adresses de début (décimales) des diverses routines de la boîte à outils :

2063-2095: place le début Basic en 16384.
2096-2112: commutateur d'affichage graphique.
2122-2147: efface l'écran graphique.
2148-2179: couleur du fond d'écran.
2181-2271: fonction POINT.
2272-2237: tableau des débuts de ligne.
2340-2480: fonction LINE.
2481-2734: fonction CIRCLE.
2735-2812: fonction ARC.
4200-4277: effacement de point.
4280-4296: module effacement.
4297-4313: retour au mode impression.
4315-4344: retour au mode texte.

M. Monsch

```

100 REM *****
110 REM * DEMONSTRATION *
120 REM *****
130 PRINT "CARTE DE FRANCE"
140 FOR I=0 TO 1500:NEXT
150 G=.3:H=.2:C=127*G:D=100*H
160 SYS(2095):SYS(2122):SYS(2148)
170 POKE8000,512*G:POKE8002,77*G:SYS(2180)
180 FORX=1TO129:READ A,B
190 POKE8001,A*G:POKE8003,B*H:SYS(2340)
200 NEXT
210 DATA66,94,459,153,467,155,402,205,398,223,408,230,387,246,377,240,353
220 DATA240,353,231,348,234,340,223,344,216,340,198,323,205,304,199,300,217
230 DATA311,223,311,243,316,246,320,299,326,305,316,311,305,304,296,315
240 DATA288,298,274,311,258,280,233,280,233,300,177,305,175,316,202,313,205
250 DATA327,180,327,180,336,194,336,194,336,198,345,170,351,180,365,198,357
260 DATA197,375,206,375,218,380,261,416,270,413,275,421,287,439,300,433
270 DATA305,445,290,450,297,450,304,457,293,480,302,510,332,538,335,567
280 DATA328,577,328,594,347,597,373,662,334,621,328,686,335,690,323,703
290 DATA315,790,300,808,316,831,316,851,328,840,363,875,377,867,389,884
300 DATA417,883,424,869,471,896,471,905,484,916,540,901,535,861,590,796
310 DATA600,802,627,812,650,822,663,839,691,831,709,820,724,802,766,763
320 DATA757,751,771,748,766,732,725,713,715,697,725,678,709,662,709,644
330 DATA730,615,713,585,725,561,705,545,705,525,695,520,672,544,664,532
340 DATA674,507,693,483,695,467,717,422,734,422,742,404,738,346,762,276
350 DATA674,264,742,270,730,255,696,249,652,223,635,229,611,194,611,176
360 DATA603,164,594,178,574,176,574,153,533,112,523,110,512,93,512,77
370 END
READY.

```

```

100 REM *****
110 REM * ROUTINES H.R POUR C.64 * (C) M.MONSCH & L'OI *
120 REM *****
130 REM ----- PROGRAMME 1: CHARGEUR BASIC POUR LES ROUTINES -----
140 PRINT "PATIENTEZ SVP: TRAVAIL EN COURS..."
150 A=0:FOR I=0 TO 1500 STEP 100
160 S=INT(SIN(.04*I*27)+.127)+INT(COS(.04*I*27))
170 IF S<0 THEN S=ABS(S)+128
180 IF S<0 THEN S=ABS(S)+128
190 POKE 2816+H+C:POKE 9584+A,S
200 A=A+1:NEXT
210 FOR I=2048 TO 2812:READ A:POKE I,A:T=T+A:NEXT
220 IF T<77796 THEN PRINT "ERREUR EN DATA BLOC 1":T:STOP
230 DATA 0,13,8,10,0,158,40,50
240 DATA 48,54,51,41,0,0,0,169
250 DATA 64,133,44,133,46,133,48,133
260 DATA 66,169,0,133,43,133,65,169
270 DATA 2,133,45,133,47,169,0,160
280 DATA 0,153,255,63,208,208,250,96
290 DATA 169,32,13,17,208,141,17,208
300 DATA 169,8,13,24,208,141,24,208
310 DATA 96,0,0,0,0,0,0,0
320 DATA 0,0,169,0,133,251,169,32
330 DATA 133,252,160,0,169,0,145,251
340 DATA 200,208,249,230,252,165,252,201
350 DATA 64,208,239,96,169,0,133,251
360 DATA 169,4,133,252,160,0,169,5
370 DATA 145,251,200,208,2,230,252,166
380 DATA 252,224,7,208,241,166,251,192
390 DATA 233,208,235,96,173,66,31,74
400 DATA 74,74,170,189,224,8,141,69
410 DATA 31,139,1,9,141,70,31,169
420 DATA 246,45,64,31,32,208,8,169
430 DATA 7,45,65,31,32,208,8,173
440 DATA 69,31,141,202,8,141,205,8
450 DATA 173,70,31,141,203,8,141,206
460 DATA 8,169,7,45,64,31,170,232
470 DATA 169,128,202,240,4,74,202,208
480 DATA 252,13,196,53,141,196,53,96
490 DATA 24,109,69,31,141,69,31,169
500 DATA 0,109,70,31,141,70,31,96
510 DATA 32,96,160,224,32,96,160,224
520 DATA 32,96,160,224,32,96,160,224
530 DATA 32,96,160,224,32,96,160,224
540 DATA 32,32,32,32,32,32,32,32
550 DATA 0,32,33,34,35,37,38,39
560 DATA 40,42,43,44,45,47,48,49
570 DATA 50,52,53,54,55,57,58,59
580 DATA 60,62,62,62,62,62,62,62
590 DATA 62,62,0,0,169,238,141,128
600 DATA 9,169,238,141,143,9,56,173
610 DATA 65,31,237,64,31,141,164,31
620 DATA 176,10,169,206,141,128,9,32
630 DATA 239,16,234,234,56,173,67,31
640 DATA 237,66,31,141,165,31,176,10
650 DATA 169,206,141,143,9,32,248,16
660 DATA 234,234,173,64,31,141,166,31
670 DATA 173,66,31,141,167,31,173,164
680 DATA 31,141,168,31,173,165,31,141
690 DATA 169,31,160,0,24,173,164,31
700 DATA 109,168,31,141,168,31,144,3
710 DATA 206,166,31,24,173,165,31,109
720 DATA 169,31,141,169,31,144,3,238
730 DATA 167,31,173,166,31,141,64,31
740 DATA 173,167,31,141,66,31,32,132
750 DATA 8,200,208,208,173,65,31,141
760 DATA 64,31,173,67,31,141,66,31
770 DATA 96,169,0,141,169,31,162,8
780 DATA 10,46,169,31,14,167,31,144

```

```

790 DATA 9,24,109,166,31,144,3,238
800 DATA 169,31,202,208,235,141,168,31
810 DATA 96,169,0,141,78,31,141,79
820 DATA 31,169,2,141,80,31,169,88
830 DATA 141,81,31,169,0,133,251,133
840 DATA 253,169,14,133,252,169,11,133
850 DATA 254,169,24,141,69,10,141,125
860 DATA 10,169,109,141,70,10,141,126
870 DATA 10,24,173,78,31,101,252,133
880 DATA 252,24,173,78,31,101,254,133
890 DATA 254,172,79,31,177,253,141,164
900 DATA 31,16,18,169,56,141,69,10
910 DATA 169,237,141,70,10,173,164,31
920 DATA 41,127,141,164,31,173,164,31
930 DATA 141,166,31,173,76,31,141,167
940 DATA 31,32,177,9,14,168,31,46
950 DATA 169,31,173,74,31,24,109,169
960 DATA 31,141,64,31,177,251,141,164
970 DATA 31,16,18,169,56,141,125,10
980 DATA 169,237,141,126,10,173,164,31
990 DATA 41,127,141,164,31,173,164,31
1000 DATA 141,166,31,173,77,31,141,167
1010 DATA 31,32,177,9,14,168,31,46
1020 DATA 169,31,173,75,31,56,237,169
1030 DATA 31,141,66,31,32,132,8,24
1040 DATA 169,1,109,79,31,141,79,31
1050 DATA 169,0,109,78,31,141,78,31
1060 DATA 173,81,31,205,79,31,240,3
1070 DATA 76,227,9,173,80,31,205,78
1080 DATA 31,240,3,76,227,9,96,169
1090 DATA 0,141,78,31,141,80,31,173
1100 DATA 79,31,141,79,31,46,78,31
1110 DATA 109,79,31,141,79,31,169,0
1120 DATA 109,78,31,141,78,31,173,81
1130 DATA 31,14,81,31,46,80,31,109
1140 DATA 81,31,141,81,31,169,0,109
1150 DATA 80,31,141,80,31,76,227,9
1160 DATA 173,164,31,141,78,31,173,167
1170 DATA 31,141,81,31,173,166,31,141
1180 DATA 80,31,76,227,9
1190
1200 FOR I=4200 TO 4353:READ A:POKE I,A:T=T+A:NEXT
1210 IF T<94835 THEN PRINT "ERREUR EN DATA BLOC 2":T:STOP
1220 DATA 173,66,31,74,74,74,170,189
1230 DATA 224,8,141,69,31,139,1,9
1240 DATA 141,70,31,169,248,45,64,31
1250 DATA 32,208,8,169,7,45,66,31
1260 DATA 32,208,8,173,69,31,141,176
1270 DATA 16,141,179,16,173,70,31,141
1280 DATA 177,16,141,180,16,169,7,45
1290 DATA 64,31,170,232,169,128,202,240
1300 DATA 4,74,202,208,252,73,255,45
1310 DATA 39,62,141,39,62,96,0,0
1320 DATA 169,104,141,159,9,141,133,10
1330 DATA 169,16,141,160,9,141,134,10
1340 DATA 96,169,132,141,159,9,141,133
1350 DATA 10,169,8,141,160,9,141,134
1360 DATA 10,96,255,169,223,45,17,208
1370 DATA 141,17,208,169,247,45,24,208
1380 DATA 141,24,208,96,0,0,0,173
1390 DATA 164,31,73,255,141,164,31,96
1400 DATA 173,165,31,73,255,141,165,31,96,0
1410
1420 PRINT "FIN DE L'IMPLANTATION EN MEMOIRE."
1430 PRINT "POUR SAUVEGARDER CE PROGRAMME SOUS FORME COMPACTE, TAPÉZ:"
1440 PRINT "POKE 43,1:POKE 44,8:POKE 45,2:POKE 46,17"
1450 PRINT "PUIS FAITES LE SAVE HABITUEL."
1460 END

```



# ALENVER

## UN PROGRAMME RENVERSANT

**En ces premiers jours de printemps, les Commodore 64 sont pris d'une douce folie. Au détour d'un couloir, j'aperçois un C 64 à l'envers. Dans un bureau tout proche, je vois un 64 à l'endroit qui travaille à l'envers. Ne pouvant résister à ma curiosité invétérée, je m'installe au clavier et tape : **LSIT**. La clé du mystère se révèle : Il suffit de lancer le programme qui porte le nom bien choisi d'Alenver. Dès ce moment, un C 64 à l'envers travaille à l'endroit, et vice versa.**

**Un conseil, il ne vous reste qu'à mettre votre C 64 la tête en bas ou à accepter de voir fleurir les torticolis dans votre entourage.**

```

100 REM ALENVER PAR B.MICHEL AVRIL 85
110 REM POUR CBM 64 RENVERSE
120 REM
130 A1$=CHR$(144)+CHR$(147):A2$=CHR$(17):A3$=CHR$(147)+CHR$(154)
140 A7$=CHR$(29):A9$=CHR$(157):A4$=CHR$(145):A5$=""+"A9$
150 A6$=CHR$(18)+A5$+CHR$(146):A8$=""+"A7$:POKE53280,255
160 POKE 53281,255:PRINT A1$ A2$ "UN PROGRAMME RENVERSANT"
170 PRINT A2$ A2$ A2$ A2$ A2$ "(UN INSTANT DE PATIENCE ET VOUS "
180 PRINT A2$"SEREZ TOUT RETOURNE !!)"
190 PRINT A2$A2$A2$ "POUSSER RETURN " : INPUT A$:PRINT A3$A2$A2$A2$A2$
200 FOR Z=4096 TO 4150:READ A:POKE Z,A:C=C+A:NEXT:IF C<>6542 THEN END
210 C(0)=1:G=8192:FOR Z=1TO7:C(Z)=2*C(Z-1):NEXT:SYS4096:POKE53272,24
220 FOR Z=G TO63*8+G STEP8:FOR J=0TO7:A(J)=PEEK(Z+J):B(J)=0:PRINT".":
230 FOR K=0TO7:B(J)=B(J)+(C((7-K)))*(A(J)AND(C(K)))/(C(K)):NEXT K,J
240 FOR J=0 TO 7:POKE 7+Z-J,B(J):NEXT J,Z:PRINT A3$:FOR Z=1 TO 20
250 PRINT A2$:NEXT:PRINT" EERF SETYB CISAB 11983 METSYS MAR K46"
260 PRINT A2$ " **** 2V CISAB 46 ERODOMMOC ****"A4$A4$A4$A4$A4$A4$
270 POKE53281,246:POKE53280,254:PRINT SPC(34):"YDAER":POKE198,0
280 PRINT CHR$(19):FORI=0TO16:PRINT A2$:NEXT:PRINT SPC(39):POKE204,0:F=1
290 F=-F:GETA$:A=ASC(A$+CHR$(0)):POKE257,255:POKE204,255:FOR Z=1 TO 80:NEXT
300 IF(A=0)AND(F=-1)THEN PRINT A6$:GOTO 290
310 IF(A=0)AND(F=1)THEN PRINT A5$:GOTO 290
320 IF A=13THEN PRINT "A4$A4$:PRINT A9$:POKE204,0:GOTO290
330 IF A=34 THEN A$="" : GOTO380
340 IFA$=A4$ THEN A$=A8$+A2$:GOTO380
350 IFA$=A2$ THEN A$=A8$+A4$:GOTO380
360 IFA$=A7$ THEN A$="" : GOTO380
370 IFA$=A9$ THEN A$=A8$+A7$
380 PRINT A$A9$A9$:POKE204,0:GOTO290
390 DATA 169,0,141,14,220,169,51,133,1,32,22,16,169,55,133,1,169,1,141,14
400 DATA 220,96,160,0,132,98,132,100,162,32,134,101,162,208,134,99,162,16
410 DATA 160,0,177,98,145,100,200,208,249,230,99,230,101,202,208,240,96

```

READY.

**ALENVER LSIT**

Benoît Michel

**VIDE SHOP**

251, bd Raspail, 75014 Paris. M° Raspail. Tél: 321.54.45  
50, rue de Richelieu, 75001 PARIS. Tél: 296.93.95  
Métro Palais-Royal. Du lundi au samedi de 9h30 à 19h

### tout, tout pour le CBM 64 !!!

- Lecteur de cassette
- Unité de disquettes 1541
- Crayon optique
- Quick data drive
- Clavier musical
- Librairie complète

et plus de 200 logiciels  
en cassette, cartouche ou disquette...

*l'espace le plus  
micro de Paris !*

Je désire recevoir gratuitement et sans engagement de ma part votre documentation sur les périphériques et logiciels de la gamme Commodore.

NOM \_\_\_\_\_

PRENOM \_\_\_\_\_

ADRESSE \_\_\_\_\_

VILLE \_\_\_\_\_

CODE POSTAL \_\_\_\_\_

Je joins 3 timbres à 2,10 F pour frais d'envoi.

# SEDERMI

## DES CARTES ET DES LOGICIELS POUR VOTRE 64

### EXT 64 BUS 4 S

Carte-mère 4 connecteurs  
Permet d'avoir 4 cartouches simultanément connectées et activables à volonté. Evite les démontage et la fatigue du connecteur.

### EXT 64 PIA 2

Carte rajoutant 40 entrées-sorties à votre 64 (idéal pour commander des appareils domestiques, trains électriques, etc.)

### Disquette EDASM 64

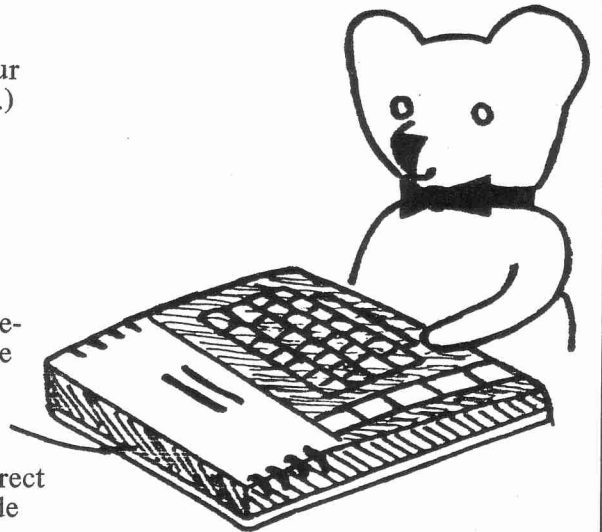
L'éditeur-assembleur universel pour 64. La version compilée est très rapide.

### Disquette TELECRAN 64

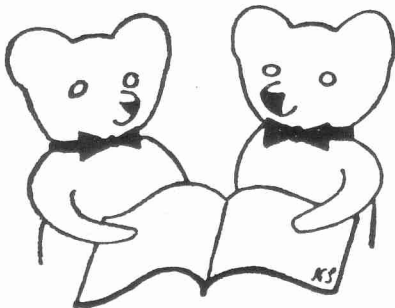
Le jeu bien connu, amélioré. Permet tout dessin haute-résolution sur l'écran, avec sauvegarde. La version compilée est spectaculairement rapide.

### Disquette BD 64

Un système de gestion, de gestion de données à accès direct sur 1541 très simple d'emploi tout en ayant des fonctions de recherche et tri élaborées.



## LA COMMODE



### LA REVUE DE REFERENCE POUR LES COMMODORE

Il ne reste plus que quelques nos 9, 10 et 11

no 9  
Ficelles de La Commode  
Elargir l'affichage  
Fonction APPEND 64  
Fonction DRAW 64, VIC  
Auto RUN  
Adresses Stratégiques  
Cours Assembleur 1

no 10  
Découverte de FORTH  
Extensions BASIC 64  
Clavier musical  
Programmathèque  
Tracé de fonctions  
Le son du 64  
Cours Assembleur 2

no 11  
La pendule magique  
Les notes sur 64  
Défilement latéral  
Magnéto ordinaire  
Sauver et Charger une zone mémoire  
Hardcopy Hte Rés.

### BON DE COMMANDE

A adresser avec le règlement à  
SEDERMI - 29, rue Vicq-d'Azir - 75010 PARIS

Nom : .....  
Adresse : .....  
.....  
.....

..... La Commode n.9 (45 F)  
..... La Commode n.10 (45 F)  
..... La Commode n.11 (45 F)  
..... EXT 64 BUS 4 S (530 F)  
..... EXT 64 PIA 2 (730 F)  
..... EDASM 64 (220 F)  
..... EDASM 64 compilé (280 F)  
..... TELECRAN 64 (140 F)  
..... TELECRAN 64 compilé (190 F)  
..... BD 64 (310 F)

règlement par :  
 CCP  CB  mandat

signature :

Sur demande, les logiciels peuvent être fournis sur cassette.  
Existe aussi pour VIC :  
EXT VIC BUS 4,  
EXT VIC PIA, EDASM, TELECRAN



**UN PRIVE  
ENQUÊTE SUR  
VOTRE BASIC**



**AUTOPSIE  
D'UNE  
MEMOIRE VIVE**

Humphrey Bogart dans le rôle de Philip Microwe, obscur détective privé New-Yorkais, Laureen Bacall dans celui de Mimi Flytow, sa blonde secrétaire, et, en vedette, vos programmes Basic examinés au microscope au cours d'une enquête complètement folle où vous apprendrez non seulement à comprendre votre Basic, mais aussi à le trafiquer, ce qui promet à vos confrères en informatique des surprises dont ils ne sont pas près de se remettre.

# BASIC-LISTE

## PAS D'OCTET POUR BERNIE SOAP

**Où l'on découvre la folle poursuite de Philip Microwe, Bernie Soap et Tony Jennet à la recherche de la recette du Basic-Liste laqué sauce Commodore.**

**V**ous deux, vous allez rester là un bon bout de temps. Moi, je me tire en Floride !

Accrochés au porte-manteau, mon feutre et mon trench-coat ne répondirent pas. Le contraire m'eût étonné, d'ailleurs. Cette affaire de l'Octet Perdu (1) m'avait rapporté de quoi payer mon ardoise chez l'Italien de Fabian Square, et je comptais bien m'offrir quelques semaines de vacances et plaquer enfin mon bureau crasseux de la 54<sup>e</sup> Avenue. Le bourbon du vieux Sam, en face, n'était pas de première qualité et vous laissait la langue comme une queue de castor, mais comme de toutes façons, je ne l'avais pas encore payé, il n'y avait rien à dire. J'allongeai la main vers le tiroir marqué « confidentiel » pour en extraire la dernière bouteille lorsque le téléphone sonna.

– Philip Microwe ? Bonsoir, mon nom est Tony Jennet et vous allez vous charger d'une affaire pour moi.

La voix était étrangement sourde et calme, mais avec un je ne sais quoi de menaçant : un bloc de feutre gris bourré de lames de rasoir. Le nom ne m'était d'ailleurs pas inconnu : fabrication clandestine d'octets, distilleries de ROM, trafic de Qwerty, Tony Jennet était certainement un homme discret, mais puissant ; je n'allais pas tarder à m'en apercevoir.

– Je regrette, l'agence est fermée pour congés annuels.

– Veuillez m'excuser si quoi que ce soit dans mes propos a pu vous laisser entendre que vous aviez le choix, monsieur Microwe. Des hommes à moi passeront vous prendre.

– Et si je refuse ?

Pour toute réponse, j'eus droit au bip bip d'une ligne téléphonique coupée. Au même moment, deux hommes firent irruption dans mon bureau. Le plus petit mesurait un bon mètre quatre-vingt quinze et avait pu passer par la porte en se mettant de profil. L'autre déposa soigneusement dans un coin ce qui restait de la porte en question et je n'eus que le temps de voir son poing grossir en direction de mon menton.

**« Une rame de métro venait de s'égarer sur mes gencives »**

Quand je revins à moi, j'eus l'impression très nette qu'une rame de métro venait de s'égarer sur mes gencives. Dans la pénombre d'un entrepôt désaffecté,

trois silhouettes étaient debout en face de moi. De celle du milieu me parvint la voix du type au téléphone :

– Monsieur Microwe, votre travail pour retrouver l'octet perdu par Clive Sinclair était surprenant, quoiqu'un peu long (1). Vous avez quinze jours pour retrouver celui de Jim Commodore. Voici trois dollars pour vos premiers frais, et je vous recommande de vous débarrasser au plus vite de ce ridicule bandage à la mâchoire. Vous pourrez me joindre en téléphonant au *Soliqueen's Club*, on m'y connaît un peu... Mais ne tardez pas trop. Je suis d'une mansuétude extrême tant qu'on ne me fait pas attendre.

Un geste, et les deux anges gardiens me reconduisirent sur le trottoir, avec des délicatesses d'éboueur.

Trois dollars ! juste de quoi offrir un nettoyage catégorie 2 à ma chemise un brin frippée par la force des arguments de ces messieurs. Je n'avais plus qu'à aller voir mon copain Li Ho Peek, fils du Ciel et de la mère Denis, blanchisseurs à Chinatown, en face de l'église.

– Salut, As-de-Peek, dis-je finement en entrant dans la sombre mais méticuleusement propre boutique du Chinois. Il me faudrait un Catégorie 2 pour ma chemise, et quelques renseignements pour moi.

### DEMANDEZ A L'INTERIEUR CE QUE VOUS NE VOYEZ PAS A L'ETALAGE

- \* Comprendre comment l'ordinateur voit les nombres.
- \* Savoir juste ce qu'il faut d'hexadécimal, sans faire de théorie abstraite.
- \* Voir octet par octet à quoi ressemble votre programme Basic en mémoire.
- \* Tutoyer les Tokens.
- \* Utiliser la mémoire d'écran pour recopier sans fatigue les programmes de quelques lignes.
- \* Ajouter un programme Basic sur disquette ou sur cassette au programme déjà en mémoire.
- \* Renommer vos programmes Basic en sachant comment ça marche.
- \* Faire passer les programmes du C.64 sur les autres ordinateurs Commodore.
- \* Obtenir des REMs à l'écran en vidéo inversée (ou en couleur) et sur imprimante en caractères expansés.
- \* Faire un programme qui se modifie tout seul.
- \* Déguiser un programme Basic en ce qui aura tout l'air d'un programme en langage machine !



– Certainement, certainement. Mon honorable ami voudrait peut-être la lecture du Basic-liste laqué sauce Commodore ?

Les affectueuses caresses des gorilleaux de Tony Jennett ne m'avaient pas davantage coupé le souffle que le don de double vue de ce vieux lotus ridé.

– Tu... enfin, je... est-ce que..., gargaris-je.

– Que mon honorable ami Miclowe veuille bien excuser son très humble selviteul, si je commence pas laisser tomber cet accent à la... (ici, quelques caractères chinois intraduisibles), mais je lis l'*Ordinateur Individuel* comme un autre, et connaissant Tony Jennett comme je le connais, c'est-à-dire jusqu'à son petit linge, ce n'était guère difficile à prévoir. Si mon honorable etc. veut bien me suivre...

Il prit au passage la maculée liquette que je lui tendais et écarta d'une main le rideau de bambou synthétique qui menait à son arrière-boutique. Une tête de belette mitée émergeait d'un pardessus genre crasseux assis sur une chaise devant un PET haute-époque.

**Le plus calé en informatique  
à une époque où  
ça n'existait même pas**

– Weaselface ! Je te croyais au Club Méditerranée...

Le petit homme au profil fuyant n'aurait inspiré confiance à personne, ni pour ses compétences ni pour son honnêteté, mais moi, je savais que les informations qu'il m'apporterait ne manqueraient pas d'intérêt. D'abord, malgré son aspect primitif et, disons-le, peu coquet, c'était l'homme le plus calé en informatique de tous les Etats de l'Union, en pleine époque de la prohibition, alors que ça n'existait même pas, c'est vous dire ! Et puis, il se rappelait sans doute que si les flics n'avaient jamais su qui avait procuré à Johnny Luck Goode la combinaison du coffre de la Federal Bank le jour du casse historique, c'était que grand-papa Microwe m'avait répété quand j'étais môme : « Fiston, tant que tu ne sais rien, cause autant que tu veux. Mais si tu sais quelque chose, alors ferme-la. » Et j'ai toujours suivi le conseil de grand-papa Microwe. Seulement, pour une stupide histoire de rafle dans un bar de la 42<sup>e</sup> Rue, Weaselface avait trinqué pour un autre, et s'était retrouvé en villégiature, bien au frais, bien à l'ombre...

– J'en avais un peu marre des G.O. d'Alcatraz, Philip, des copains sont venus me chercher. Pourtant, là-bas, je n'ai pas perdu mon temps. Regarde.

Sur l'écran du vieux PET se succédaient des pages de texte et des tableaux clignotants.

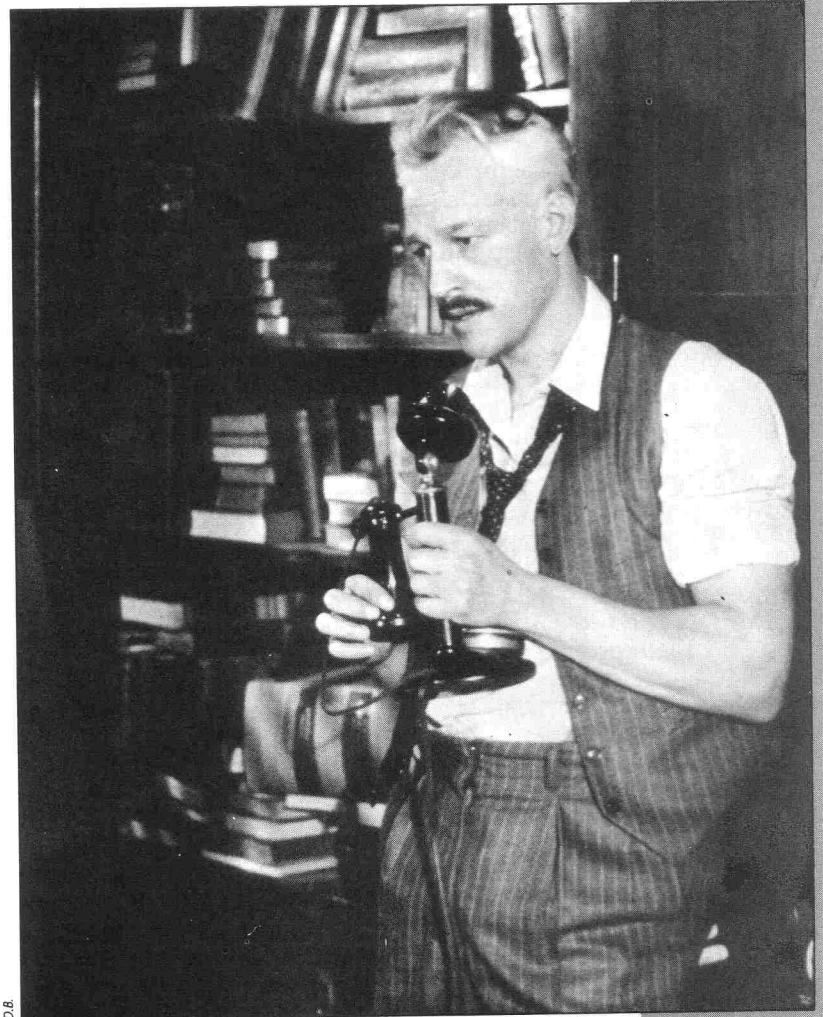
– Je m'occupais du club informatique, poursuivait Weaselface, et j'ai... recoupé quelques petits programmes. Je crois que celui-ci t'intéressera.

– Qu'est-ce que c'est que ce truc-là ?

– Il s'agit là d'un programme hautement confidentiel destiné à expliquer comment les ordinateurs de chez Commodore, les PET, les CBM, le VIC 20 et le C 64 stockent leurs programmes BASIC.

– Basic-liste ! murmurai-je.

– Tous les appareils de la marque présentent des caractéristiques communes. Regarde bien. Un des intérêts du programme, c'est qu'il tourne aussi bien sur ce vieux PET que sur les CBM 3000, 4000 et 8000 ou sur le 64. Les lignes 50120 à 50150 commencent par l'instruction REM. On efface le mot REM devant la ligne correspondant à l'ordinateur sur le-



quel le programme vient d'être chargé, et les adresses sont fixées une fois pour toutes.

– Et si on oublie de le faire ?

– Le programme liste les premières lignes, te dit ce qu'il faut faire et s'arrête.

– Mais ces caractéristiques communes, c'est quoi ?

Dans la pénombre verte, le Chinois s'affairait sur sa lessive. Baladant sur l'écran un index noirâtre, Weaselface expliqua :

– Prima, avant le programme proprement dit, on trouve toujours un zéro. Secundo, les deux premiers octets de chaque ligne représentent l'adresse de début de la ligne suivante. Tertio, les deux octets suivants représentent le numéro de la ligne. Quarto, chaque ligne de programme est terminée en mémoire par un zéro.

– Voilà qui ressemble à peu de chose près à ce qui se passe chez Sir Clive... Mais ce zéro d'avant le programme, tu connais son adresse ?

Weaselface sortit de sa poche un paquet de Pull Moll bout filtre, y puisa la dernière, la défripa, l'alluma posément avant de griffonner sur l'emballage :

PET/CBM .....	1024 (\$0400)
VIC 20 de base .....	4096 (\$1000)
VIC 20 + 3 K .....	1024 (\$0400)
VIC 20 + 8 ou 16 K .....	4068 (\$1200)
C.64 .....	2048 (\$0800)

(1) « Le mystère de l'Octet Perdu » in l'*Ordinateur Individuel* n° 61 à 64.

Il y eut d'abord cette voix : un bloc de feutre gris bourré de lames de rasoir. Adieu la Californie, mes vacances étaient foutues...

Bigre, c'est vrai qu'il n'avait pas perdu son temps, le cradingue. Je rangeai soigneusement le précieux document dans mon étui à cigarettes en cuir, autre héritage du vieux Microwe. Li Ho Peek en avait terminé avec ma chemise. Je pus enfin dissimuler aux regards jaloux mes puissants deltoïdes. Weasel-face me fit asseoir sur une caisse de bourbon vide qui n'avait pas dû rapporter grand-chose au fisc. Il fourragea dans son pardessus et étala devant lui une liasse de papiers grassex.

**Un programme qui tourne sur PET, CBM 3000, 4000 et 8000 ou sur le C.64**

— Nous pourrions en tirer quelques petites conclusions plus tard. Mais en attendant, jetons un œil sur le programme.

**LES MOTS CLES  
(OU TOKENS) DE BASIC**

Lorsqu'un programme Basic est en mémoire, les mots clés ne sont pas stockés en toutes lettres. P R I N T, par exemple, n'est pas représenté par les cinq lettres, mais par un seul octet contenant la valeur 153. On remarque que ces TOKENS ont tous une valeur supérieure à 128. A l'exécution, lorsque Basic rencontre un octet de valeur supérieure à 128, il cherche si des guillemets sont ouverts. Dans l'affirmative, il s'agit d'un caractère shifté sans plus. Sinon, il interprète cet octet comme une instruction. C'est ce qui fait que les majuscules sont inutilisables en REM, à moins d'être comprises entre guillemets.

Voici la liste des instructions BASIC avec leurs codes. Les codes compris entre 204 et 218 sont propres au BASIC 4.0.

128 END	158 SYS	188 LOG
129 FOR	159 OPEN	189 EXP
130 NEXT	160 CLOSE	190 COS
131 DATA	161 GET	191 SIN
132 INPUT#	162 NEW	192 TAN
133 INPUT	163 TAB(	193 ATN
134 DIM	164 TO	194 PEEK
135 READ	165 FN	195 LEN
136 LET	166 SPC(	196 STR\$
137 GOTO	167 THEN	197 VAL
138 RUN	168 NOT	198 ASC
139 IF	169 STEP	199 CHR\$
140 RESTORE	170 +	200 LEFT\$
141 GOSUB	171 -	201 RIGHT\$
142 RETURN	172 *	202 MID\$
143 REM	173 /	203 GO
144 STOP	174 ^	204 CONCAT
145 ON	175 AND	205 DOPEN
146 WAIT	176 OR	206 DCLOSE
147 LOAD	177 =>	207 RECORD
148 SAVE	178 =	208 HEADER
149 VERIFY	179 =<	209 COLLECT
150 DEF	180 SGN	210 BACKUP
151 POKE	181 INT	211 COPY
152 PRINT #	182 ABS	212 APPEND
153 PRINT	183 USR	213 DSAVE
154 CONT	184 FRE	214 DLOAD
155 LIST	185 POS	215 CATALOG
156 CLR	186 SQR	216 RENAME
157 CMD	187 RND	217 SCRATCH
		218 DIRECTORY

C'est un programme en trois parties :

1. Explication de quelques principes de base (c'est le moment ou jamais) de l'hexadécimal et des nombres sur deux octets.
2. Présentation de la manière dont est emmagasiné en mémoire le début d'un programme Basic.
3. Analyse d'un programme entier.

Tu me suis ? Dans un premier temps, donc, après quatre secondes d'initialisation, six pages-écran d'explications rappellent ce qu'il faut connaître de l'hexadécimal, et surtout des notions de poids fort et de poids faible. Le point de vue est purement pratique et n'ajoute rien à ce qui a été déjà dit et redit, mais le fait que cela apparaisse à l'écran permet de rendre le programme compréhensible sans documentation. Tu vois ce que je veux dire... une supposition que quelqu'un veuille faire passer des copies du programme dans les clubs informatiques clandestins, c'est plus simple. A la fin de ces explications, l'utilisateur se voit offrir la possibilité de revoir le premier chapitre ; c'est pour cela que les exemples utilisés font appel aux fonctions aléatoires de l'ordinateur : ils seront différents chaque fois.

— Hun hun, fis-je parce que le gars chargé de la traduction pour la version française doublée n'avait rien trouvé de plus expressif à me faire dire. Weasel-face, ravi d'étaler sa science, continua :

— La seconde partie fait apparaître, dans la moitié supérieure de l'écran, la mémoire du début de Basic : à gauche, les adresses, puis, sur trois colonnes, le contenu des cases-mémoire.

— Tiens ! Pourquoi seulement trois colonnes ? Le moindre moniteur en offre huit, et même sur le VIC 20, on peut en avoir cinq ?

— C'est qu'on fait dans le luxe, privé à la manque, et que cette partie-là du programme est bilingue : chaque nombre apparaît à la fois en hexa et en décimal.

— Au début, ça facilite bien les choses, c'est vrai, admis-je. Mettons que je n'aie rien dit.

Il jubilait.

— La moitié inférieure de l'écran est consacrée au commentaire des octets critiques : liens d'adresses, numéros de lignes et fin de ligne. Une boucle de temporisation permet aux commentaires d'apparaître un par un, cependant que les octets désignés clignotent gaiement à l'écran. Là encore, l'utilisateur peut revoir le chapitre entier s'il le désire.

— Il faut que je te fasse un petit sifflement admiratif ?

— Passe la main, tu veux ? Le troisième temps du programme fait simplement l'analyse, octet par octet, du contenu d'un programme Basic. Ce peut être le programme de démonstration lui-même, mais comme il est numéroté à partir de 50000, cela permet soit de frapper au clavier un programme expérimental et de voir à quoi il ressemble, vu de l'intérieur, soit, pour ceux qui disposent d'un utilitaire MERGE ou APPEND (en soft ou en MEM, EDEX, SIMONS'BASIC ou ARROW, par exemple) de charger un programme sur cassette ou disquette et de le désosser posément (2).

**A gauche de l'écran, en vidéo inversée, défilaient les numéros des cases-mémoire**

Il me ramena vers l'écran glauque, et d'un GOTO rapide, fit sauter le programme directement à sa





D.B.

troisième phase. A gauche de l'écran, en vidéo inversé, défilait les numéros des cases-mémoire, puis leur contenu en décimal ; à droite, ce contenu s'affichait en clair selon sa position dans la ligne : adresse de la ligne suivante, numéro de ligne, code-opération Basic (TOKEN) ou caractère ASCII.

– On peut savoir comment il fait pour les reconnaître ? Ton programme est physionomiste ?

– Pour reconnaître un TOKEN d'un ASCII, c'est une question de QUOTE, dit-il négligemment.

Je répondis par le sourire crispé et vaguement inquiet d'un flic (américain) devant une finesse.

– Une question de guillemets, monsieur le détective. Pour distinguer les deux, il y a une variable, la variable 'q', qui t'indique si tu es ou non en mode guillemets (tu as déjà essayé de mettre en REM des caractères shiftés sans guillemets autour ? Non ? Eh bien essaie un peu pour voir, tu auras du spectacle...). Pendant cette partie du programme, il suffit d'appuyer sur une touche pour arrêter le défilement. Tu appuies encore pour le reprendre. Le programme s'arrête tout seul s'il tombe sur le triple zéro fatidique qui marque la fin de tout programme Basic qui se respecte (ou si tu préfères, quand l'adresse de la prochaine ligne est zéro), mais comme la touche STOP n'est pas inhibée, tu peux aussi t'en servir pour sortir gracieusement avant la fin si tu veux.

Je sortis mon briquet, allumai un chewing-gum – c'est dur d'arrêter de fumer – et me renversai en arrière méditativement. Ainsi, Weaselface avait fait, sans le savoir, tout mon travail. S'il me donnait ces documents, je n'avais plus qu'à les apporter à Tony Jennett, empocher la prime, et je pouvais même offrir aussi des vacances au crasseux, mais cette fois-ci, au soleil.

– Que le très honorable privé veuille bien me pardonner si j'ai l'audace de troubler sa méditation, mais je viens d'entendre un léger bruit dont tout

laisse à penser qu'il s'agit du cran de sûreté d'un Beretta calibre 38 modèle 1929 production fin octobre, usine de Detroit, et que, par voie de conséquence, notre intérêt commun serait de nous imposer une position allongée ventrale et dare-dare, parce que ça va...

### Le PET vola en éclats... mes vacances semblaient compromises

Il y eut en effet un instant de léger désordre ; le PET vola en éclats ainsi que les vitres de l'étroit vasistas. Le bruit disgracieux d'une fusillade perturbait la sérénité de l'arrière-boutique, et, lorsque, méticuleux, j'eus ôté de mon trench-coat les derniers fragments de plafond, je dus constater que mes vacances semblaient compromises. J'avoue avoir hésité un instant entre des œillets et des roses pour la tombe de Weaselface. Quant à Li Ho Peek, il était au bord de l'irritation :

– Ces fils de blatte m'ont interrompu et ils ont occis notre ami. C'est éminemment mal élevé.

– Ils ont aussi emporté son programme, soupirai-je alors que, dehors, démarrait une voiture dans un crissement de pneus à vous faire frémir les caries.

– Buick 12 litres 6 dernier modèle à carburateur double corps, équipée de pneus Michigan à carcasse inox et flancs blancs. Je distingue mal au son les carrosseries noires des rouges, mais c'est l'un des deux, murmura Li Ho Peek comme pour lui-même.

Deux mots me trottaient dans la tête : *Soliqueen's Club*. Il était peut-être temps d'y aller faire un tour...

(2) EDEX : utilitaires en MEM pour CBM 3000 - 4000 - 8000, distribution PROCEP - 529 F.

ARROW : cartouche pour C.64. Importée par RUN INFORMATIQUE, 62, rue Gérard, 75013 Paris, tél. : (1) 581.51.44.

Lorsque la Buick 12 litres 6 fit entendre son chant, Weaselface avait cessé de vivre, emportant avec lui les secrets de Basic-Liste.

# TONY JENNETT CHANGE D'ADRESSES

**Un détective privé, dans un club du même nom, découvre un programme de renumérotation qui pourrait bien lui sauver la peau.**

Une heure du matin. La 256<sup>e</sup> Rue. Un lançant solo de saxophone s'échappe d'une fenêtre au-dessus de moi. Les néons rouges des boîtes de nuit se reflètent avec une lueur sinistre sur l'asphalte, mouillé par un consciencieux accessoiriste hollywoodien. Sur le trottoir d'en face s'arrêtent d'interminables limousines noires, aussitôt vidées de leur contenu chic par des amiraux de l'U.S. Navy déguisés en portiers – à moins que ce ne soit l'inverse. Un dais rouge protège le tout, surmonté d'une enseigne au clignotement échevelé : *Soliqueen's Club*.

D'un geste précis, je lisse le bord de mon feutre mou et laisse tomber ma cigarette, vite écrasée d'un mouvement circulaire du pied gauche. Humphrey Bogart passe et me regarde faire. J'ai l'impression que ça lui servira...

Je ne savais pas si les dollars qui avaient payé la moquette étaient à base de distilleries clandestines ou de trafic d'armes, mais le maître des lieux avait bien fait les choses, et, entre deux haies de palmiers en pots, l'entrée du *Soliqueen's Club* n'annonçait pas l'établissement minable. Un portier galonné s'approche et jauge mon trench-coat. Pour lui, c'est simple, je me trompe d'adresse.

– Puis-je quelque chose pour vous ?

– Mais certainement, Monsieur le Chef de gare. Je cherche le buffet, j'ai envie d'un verre de lait.

Il se prépare déjà à faire signe au videur maison, mais, plus rapide que lui, j'ai déjà retourné le revers de mon col. Un insigne luit sous le néon.

– Excusez-moi, inspecteur, donnez-vous la peine d'entrer.

Lorsqu'à quinze ans j'avais remporté l'épreuve intercommunale de natation à la piscine de Beauties-Barn, je n'avais pas remarqué à quel point la médaille qu'on m'avait remise ressemblait à un insigne d'inspecteur de police. Je m'étais bien rattrapé depuis.

Trente mètres de gazon laineux plus loin, c'était le club. Plantes vertes et velours rouge, miroirs et dorures, lumière feutrée, ambiance tamisée, smokings et perlouzes. Dans un coin, un pianiste guimauvait du clavier. Derrière son bar, un bellâtre jouait du shaker avec le même doigté que Sonny Igoe aux maracas dans *Goodman Blues*.

Trois bourbons et quelques dollars plus tard, j'avais sensiblement augmenté mon fichier intérieur à l'article *Soliqueen's*. Le patron était un certain Bernie Soap, qui possédait également deux ou trois journaux du soir, dont l'*Ordinary Inquirer*. La salle de jeux était à l'entresol et son bureau au premier, ce qui lui permettait une surveillance personnelle. Pourtant, c'était la porte du fond à gauche qui m'intéressait le plus : tout le décor de la salle semblait destiné à détourner d'elle l'attention, et le seul type que j'avais vu en sortir avait jeté autour de lui le regard caractéristique du bonhomme qui cherche à voir si on l'a repéré.

Comment avez-vous deviné que je l'ai franchie, cette porte, et qu'après avoir parcouru des couloirs ripolinés, j'ai mis moins de cinq minutes à ouvrir une

autre porte qu'on souhaitait visiblement garder fermée ? Vous êtes vraiment fort, lecteur !

**un dé clic : la porte  
du drive s'ouvre  
sous mon index**

Une pièce nue. Un soupirail bée sur des chevilles. Le mobilier réformé du bar : des chaises et une table. Mais sur cette table, *mamma mia* ! c'est un Commodore 64, avec disquette et moniteur couleur. Ça aurait fait plaisir à feu Weaselface. Pensée émue pour toi, mon pote, on s'occupe des gars qui t'ont sulfaté. Et même pas plus tard que tout de suite : c'est ton programme qui est là. Et à en juger par les papiers épars, ils sont en train d'essayer de le comprendre. Les vaches !

Un dé clic : la porte du drive s'ouvre sous mon index et la disquette se retrouve dans ma poche. Mais puisqu'ils veulent comprendre ton programme, laisse-moi te dédicacer un petit gag posthume, vieux frère. Qu'est-ce que je pourrais bien leur offrir ?

De là-haut me parvient la rengaine assourdie du piano. L'article a abandonné son style guimauve et s'est lancé à corps perdu dans *Softwar Stomp*. *Softwar Stomp*... Un éclair jaillit. De mon poing droit fermé, je frappe violemment la paume de ma main gauche et ne peux retenir une exclamation : « Mais bon sang, mais c'est bien sûr ! » Dans le couloir, Claude Loursais passe et me regarde faire. J'ai l'impression que ça lui servira...

*Softwar*, vous savez, le roman d'André Breton et Guillaume Budé, où la guerre consiste à truffer de bogues le programme des petits copains... Ah, tu voulais la guerre, Tony Jennett, eh bien, j'ai comme l'impression que tu ne vas pas tarder à perdre une bataille !

Une supposition que j'ajoute au programme une ligne ainsi conçue :

```
53425 FOR I=0 TO
255:POKE INT(RND(1)*
255),INT(RND(1)*
255):NEXT I
```

Celle-là, elle fait mal par où elle passe. Mais il faut la planquer dans un programme numéroté de 10 en 10 et où un numéro de ligne terminé par un 5 apparaîtrait tout de suite comme un non-sens, au moins autant qu'une assiette propre dans un fast-food. Non, le seul moyen, c'est de tout renumérotter. Et ce, par quelques lignes de programme ajoutées, puis



D.B.



effacées après usage. La première chose à faire est donc d'adapter au Commodore le programme de renumérotation automatique découvert chez Sir Clive.

Résumons-nous, si un programme Basic commence à l'adresse A, PEEK (A+1) + 256\* PEEK (A+2) renvoie l'adresse du début de la ligne suivante, et le numéro de la ligne en cours est contenu dans l'ordre Poids faible/Poids fort en A+3 et A+4. De là à faire un additif renumérotatoire, il n'y a qu'un pas.

J'ai toujours dans mon portefeuille la liste d'adresses qu'avait dressée Weaselface au dos de son paquet de Pull Moll bout filtre (voir encadré « Quelques adresses utiles »). Prenons l'adresse qui suit immédiatement l'adresse indiquée : pour un 64, Weaselface m'indique 2048. Je prends 2049. C'est de cette adresse que tout part.

Histoire à la fois de gagner de la place sur les lignes et du temps dans la boucle, je paramètre la valeur 256 que je baptise B.

### Un double INPUT à message intégré et virgule médiane

L'utilisateur doit avoir le libre choix du numéro de ligne auquel commencera le programme renuméroté, et le libre choix du « pas » entre les lignes (de 1 en 1, de 5 en 5, de 10 en 10, etc.). « This is a free country, man... », disait le vieux grand-papa Microwe de sa voix un peu traînante, dans les situations les moins appréciées par les flics. Un double INPUT à message intégré et virgule médiane, et voilà le numéro de la ligne de départ rangé dans la variable D, et le pas dans la variable P.

Puisque c'est un programme qui boucle, j'avais d'abord pensé à boucler par un GOTO, mais comme les numéros de lignes changent – heureusement ! puisque c'est le but de l'opération – je risquais de me

## QUELQUES ADRESSES UTILES

	PET	CBM	VIC 20 nu	VIC 20 + 3K	VIC 20 + 8/16 K	C.64
Début de BASIC	1 024	1 024	4 096	1 024	4 608	2 048
Pointeur de début de BASIC	122	40	43	43	43	43
Mémoire-écran	32 768	32 768	7 680	7 680	4 096	1 024
Pointeur de mémoire-écran			648	648	648	648
Dernière touche enfoncée	515	151	197	197	197	197

trouver dans un fichu guêpier. Alors, on va s'en tirer avec une boucle prévue pour un nombre de tours très supérieur à ce qu'il sera, quitte à sauter en marche lorsque tout sera fait.

Si je saute A qui contient de toutes manières un zéro, la prochaine ligne commencera à une nouvelle adresse, NA, contenue en A et A + 1. Mettons-la de côté dans la variable NA, elle nous servira plus tard. Quant au numéro que je vais imposer à la première ligne, et que j'ai baptisé D, comme il sera codé sur deux octets, autant le convertir tout de suite : l'octet de poids fort contiendra la partie entière du numéro divisé par 256, et l'octet de poids faible, le reste. Pour l'octet de poids fort, cela peut s'écrire (D0 = INT (D/256)). Mais d'une part, j'ai déjà affublé 256 du surnom de B, et, d'autre part, les noms de variables terminés par le signe % désignent uniquement des nombres entiers. Double avantage, puisqu'on a une fonction de moins à écrire et à exécuter, et que les entiers sont traités plus vite que les nombres en

(Suite de notre enquête page 37)

Le Soliqueen's Club fut fermé pour back-up clandestin, trafic de Qwerty, et distillerie de ROM. Pourtant ce cabaret new-yorkais devait connaître peu après un destin fabuleux...



# A QUOI JOUE-T-ON ?

Le but essentiel de Basic-liste est de « désosser » un autre programme. Connaître, comprendre, exploiter... tel est notre devise ! Il vous sera vraiment utile si vous avez déjà frappé un autre programme. C'est pour cela que nous l'avons numéroté à partir de 50000. Avantage supplémentaire, il peut se désosser lui-même. Avant de nous lancer, examinons un programme simple, quelque chose comme :

```
100 REM TEST
110 GOTO 50000
120 PRINT « (CLR) TEST »
130 FOR I=1 TO 10
140 PRINT I
150 NEXT I
160 END
```

Une astuce pour effectuer un APPEND.

Comment créer la commande APPEND. La fonction MERGE est absente du Basic du C.64. Cependant, on peut l'obtenir très aisément, à condition, bien sûr, qu'il y ait encore de la place en mémoire.

- lire les adresses 45 et 46 indiquant l'adresse de fin du programme stocké en MEV (ce n'est pas obligatoire, mais cela indique aussi la MEV restante,
- dire au C.64 que le Basic commence seulement à cet endroit en tapant :  
POKE 43, PEEK (45)-2 puis : POKE 44, PEEK (46)  
(PEEK (45)-2, afin de supprimer les 00 indicateurs de fin de programme) ;
- taper : LOAD... (programme à charger), après avoir placé la disquette ou K7 ;
- une fois le chargement effectué, réinstaller le départ initial du Basic :  
POKE 43, 1 et POKE 44, 8.

Le C.64 opérera la fusion des deux programmes mais attention aux numéros de lignes !

Rappelons que APPEND permet d'ajouter un programme B au bout d'un programme A à condition que les numéros de ligne de B soient supérieurs aux numéros de ligne de A.

L'ordre ne sera pas rétabli (cas des GOTO et GOSUB).

S'il n'y a pas de branchement, cela fonctionnera même en désordre, le C. 64 attendant de rencontrer les « 00 » pour décider s'il est arrivé à la fin du programme.

On peut ainsi avoir (et sauvegarder) un programme commençant aux lignes 100 et suivantes, et se poursuivant aux lignes 20 et suivantes (100... y compris).

Hormis certaines applications à déterminer, il vaut mieux, de façon générale, s'efforcer d'avoir des numéros de lignes qui vont normalement en ordre croissant. Il faut pour cela que le programme à charger ait des numéros supérieurs au programme se trouvant en MEV au moment où l'on désire utiliser la commande MERGE. Vous pouvez donc démonter d'importants programmes que vous aurez déjà sur cassette ou sur disque. Mais notre propos est surtout de comprendre à quoi ressemble un programme Basic en mémoire, pour le plaisir d'abord, et puis peut-être aussi pour en tirer quelques conséquences plaisantes-z-et utiles.

## QUELQUES CONSEILS POUR FRAPPER BASIC-LISTE

Passez en minuscules en appuyant simultanément sur les touches SHIFT et Commodore. Les caractères soulignés doivent être entrés en majuscules (SHIFT).

Le signe Tilde (~) sert seulement à indiquer qu'une ligne a été coupée.

Ne le cherchez pas au clavier !

Les mots entre crochets signalent des caractères de contrôle :

(3DOWN) signifie 3 curseurs bas.

(3SPACES) signifie 3 espaces.

(2SH.SP) signifie deux espaces shiftés, mais dans ce programme, des espaces normaux conviennent.

## CARACTÈRES GRAPHIQUES :

& représente le pavé quadrillé. On l'obtient avec les touches Commodore et +

( représente le demi-pavé quadrillé horizontal. On l'obtient avec les touches Commodore et E

@ représente la barre horizontale. On l'obtient avec les touches SHIFT et \*

2 représente le raccord de tableau en T. On l'obtient avec les touches Commodore et R

≡ représente le coin carré inférieur droit. On l'obtient avec les touches Commodore et X

Avant d'exécuter le programme, supprimer le mot REM au début de la ligne 50150.





```

50990 PRINT"LE SECOND: ";
51000 PRINT X$;" = (";X1$;" X 100) + ";X2~
$:PRINT
51010 PRINT,,X1$;"00"
51020 PRINT,,"+ ";X2$
51030 PRINT,,"@@@@"
51040 PRINT,,X$
51050 PRINT"{DOWN}* QUE LA PREMIERE MOITI~
E EST DONC 100"
51060 PRINT"FOIS PLUS IMPORTANTE QUE LA S~
ECONDE."
51070 GOSUB 53420
51080 X=INT(RND(1)*61439)+4096
51090 X1=INT(X/256):X2=X-X1*256
51100 PRINT"{CLR}PRENONS AU HASARD UN NOM~
BRE HEXADÉCIMAL"
51110 PRINT"A QUATRE CHIFFRES : {RVS}$";:~
D=X:GOSUB 53080
51120 PRINT"{OFF} (=";MID$(STR$(X),2);)"
51130 PRINT"{DOWN}COUPONS-LE EN DEUX: $";~
51140 D=X1:GOSUB 53100:PRINT" ET $";:D=X2~
:GOSUB 53100:PRINT
51150 PRINT,,"(";MID$(STR$(X1),2);" ET ";~
MID$(STR$(X2),2);)"
51160 PRINT"ON REMARQUE:"
51170 PRINT"{DOWN}* QUE PUISQUE CHAQUE MO~
ITIE COMPORTE"
51180 PRINT"DEUX CHIFFRES, CHACUNE EST IN~
FERIEURE"
51190 PRINT"AU CARRE DE LA BASE (16 FOIS ~
16=256)."
51200 PRINT"{DOWN}* QUE POUR RECONSTITUER~
LA VALEUR DE CE"
51210 PRINT"NOMBRE, IL FAUT MULTIPLIER LE~
PREMIER"
51220 PRINT"PAR LE CARRE DE LA BASE ET Y ~
AJOUTER"
51230 PRINT"LE SECOND:";
51240 PRINT X;"= (";X1;"X 256 ) +";X2
51250 PRINT,,"{DOWN}$";:D=X1:GOSUB 53100:P~
RINT;"00",
51260 PRINT RIGHT$("{3SPACES}" +STR$(X1*25~
6),7)
51270 PRINT,,"+ $";:D=X2:GOSUB 53100:PRINT~
,
51280 PRINT RIGHT$("{3SPACES}+ " +STR$(X2)~
,7)
51290 PRINT,"@@@@@"," @@@@@@"
51300 PRINT,"$";:D=X:GOSUB 53080:PRINT,
51310 PRINT RIGHT$("{5SPACES}" +STR$(X),7)~
51320 PRINT"{DOWN}* QUE LA PREMIERE MOITI~
E EST DONC 256"
51330 PRINT"FOIS PLUS IMPORTANTE QUE LA S~
ECONDE."
51340 GOSUB 53420
51350 PRINT"{CLR}ON APPELLE"
51360 PRINT"{DOWN}{3SPACES}OCTET DE POIDS~
FORT"
51370 PRINT"OU OCTET LE PLUS{2SPACES}SIGN~
IFICATIF (OPS)"
51380 PRINT"OU MOST{2SPACES}SIGNIFICANT B~
YTE{6SPACES}(MSB)"
51390 PRINT"{DOWN}LA PREMIERE MOITIE DU N~
OMBRE."
51400 PRINT"{2DOWN}L'AUTRE MOITIE EST APP~
ELEE:"
51410 PRINT"{DOWN}{3SPACES}OCTET DE POIDS~
FAIBLE"
51420 PRINT"OU OCTET LE MOINS SIGNIFICATI~
F (OMS)"
51430 PRINT"OU LEAST SIGNIFICANT BYTE{6SP~
ACES}(LSB)"
51440 GOSUB 53420
51450 PRINT"{CLR}LORSQUE LE ";N$;" STOCKE~
UN NOMBRE,"
51460 PRINT"{DOWN}IL LE FAIT GENERALEMENT~
DANS L'ORDRE"

```

```

51470 PRINT"{DOWN}POIDS FAIBLE / POIDS FO~
RT (LSB/MSB)."
51480 PRINT"{2DOWN}DANS LA PRATIQUE, IL S~
UFFIT DE SE"
51490 PRINT"{DOWN}SE RAPPELER QUE SI UN N~
OMBRE EST STOCKE"
51500 PRINT"{DOWN}SUR DEUX CASES-MEMOIRE ~
SUCCESSIVES,"
51510 PRINT"{DOWN}C'EST LE CONTENU DE LA ~
{RVS}DEUXIEME"
51520 PRINT"{DOWN}QU'IL FAUT MULTIPLIER P~
AR 256."
51530 GOSUB 53420
51540 PRINT"{CLR}VOULEZ-VOUS REVOIR CES N~
OTIONS ?"
51550 PRINT,,"{DOWN}(O/N)"
51560 GET R$:IF R$<>"O" AND R$<>"N" THEN ~
51560
51570 IF R$="O" THEN 50550
51580 PRINT"{CLR}EXPLORONS LE CONTENU DE ~
LA MEMOIRE"
51590 PRINT"{DOWN}DU ";N$;" A L'ENDROIT O~
U IL MET"
51600 PRINT"{DOWN}LES PROGRAMMES BASIC, C~
'EST A DIRE"
51610 PRINT"{DOWN}A L'ADRESSE";LM;"($";:D~
=LM:GOSUB 53080:PRINT")."
51620 GOSUB 53420
51630 REM.
51640 REM -----
51650 REM LIENS
51660 REM -----
51670 REM.
51680 AA=LM:GOSUB 52560
51690 PRINT T$:PRINT"VOYONS UN PEU CE QUI~
SE PASSE EN ";MID$(STR$(PA),2);".~
"
51700 GOSUB 53420
51710 AA=PA:GOSUB 52560
51720 PRINT"LA PROCHAINE LIGNE DE BASIC C~
OMMENCE"
51730 PRINT"A L'ADRESSE ";MID$(STR$(PA),2~
);", ET AINSI DE SUITE."
51740 GOSUB 53420
51750 REM.
51760 REM -----
51770 REM NUMEROS DE LIGNES
51780 REM -----
51790 REM.
51800 AA=LM:GOSUB 52940:PRINT"REVENONS A ~
L'ADRESSE ";MID$(STR$(LM),2);"."
51810 PRINT"LES OCTETS 3 ET 4 D'UNE LIGNE~
DE BASIC"
51820 PRINT"CONTIENNENT LE NUMERO DE LIGN~
E."
51830 GOSUB 53360:A0=2:GOSUB 53290:A0=3:G~
OSUB 53290
51840 PRINT,MID$(STR$(PEEK(LM+2)),2);"+(2~
56X";MID$(STR$(PEEK(LM+3)),2);")="~
;
51850 PRINT MID$(STR$(PEEK(LM+2)+256*PEEK~
(LM+3)),2)
51860 PRINT"LE NUMERO DE LA 1ERE LIGNE ES~
T DONC";
51870 PRINT MID$(STR$(PEEK(LM+2)+256*PEEK~
(LM+3)),2):GOSUB 53360:GOSUB 53360~
51880 PA=PEEK(LM)+256*PEEK(LM+1):PA=PA+2:~
A0=PA-LM:GOSUB 53290
51890 PA=PA+1:A0=PA-LM:GOSUB 53290
51900 PRINT"LE NUMERO DE LA LIGNE SUIVANT~
E EST"
51910 PRINT,MID$(STR$(PEEK(PA-1)),2);"+(2~
56X";MID$(STR$(PEEK(PA)),2);")=";
51920 PRINT MID$(STR$(PEEK(PA-1)+256*PEEK~
(PA)),2)
51930 PRINT"{DOWN}...ET AINSI DE SUITE."
51940 GOSUB 53420
51950 REM.
51960 REM -----

```



```

51970 REM TEXTE
51980 REM -----
51990 REM.
52000 PRINT"{CLR}{3SPACES}ENTRE LE NUMERO~
DE LIGNE ET LE ZERO"
52010 PRINT"FINAL, LE TEXTE DE LA LIGNE."~
52020 PRINT"{3SPACES}LES MOTS-CLES DE BAS~
IC NE FIGURENT"
52030 PRINT"PAS EN TOUTES LETTRES, MAIS S~
OUS FORME"
52040 PRINT"DE CODES (OU TOKENS), PAR EXE~
MPLE:{DOWN}"
52050 FOR I=128 TO 153:PRINT I,C$(I),:NEX~
T:PRINT",,","... ETC."
52060 PRINT"{DOWN}LE RESTE EST REPRESENTE~
EN CODE ASCII."
52070 GOSUB 53420
52080 PRINT"{CLR}VOULEZ-VOUS REVOIR CES N~
OTIONS ?"
52090 PRINT",,","(O/N)
52100 GET R$:IF R$<>"O" AND R$<>"N" THEN
52100
52110 IF R$="O" THEN 51680
52120 PRINT"{CLR}VOICI DONC LE CONTENU DU~
PROGRAMME BASIC"
52130 PRINT"DEPUIS SON DEBUT."
52140 PRINT"{2DOWN}POUR ARRETER LE DEFILE~
MENT,"
52150 PRINT"{DOWN}APPUYER SUR UNE TOUCHE.~
"
52160 PRINT"{2DOWN}POUR REPREDRE LE DEFI~
LEMENT,"
52170 PRINT"{DOWN}APPUYER SUR UNE TOUCHE.~
"
52180 PRINT"{2DOWN}POUR ARRETER LE PROGRA~
MME,"
52190 PRINT"{DOWN}APPUYER SUR STOP."
52200 GOSUB 53420
52210 PRINT"{CLR}";
52220 REM.
52230 REM =====
52240 REM EXPLORATION DU PROGRAMME
52250 REM =====
52260 REM.
52270 AA=LM
52280 REM ----- LIEN
52290 PA=PEEK(AA)+256*PEEK(AA+1):Q=0
52300 IF PA=0 THEN PRINT"{DOWN}FIN DU PRO~
GRAMME":GOTO 52470
52310 PRINT"{RVS}";RIGHT$("{3SPACES}"+STR~
$(AA),5);"{OFF}2 PROCHAINE LIGNE :~
";PA
52320 PRINT"{RVS}";RIGHT$("{3SPACES}"+STR~
$(AA+1),5);"{OFF}="
52330 REM ----- NO. DE LIGNE
52340 PRINT"{RVS}";RIGHT$("{3SPACES}"+STR~
$(AA+2),5);"{OFF}2 NUMERO DE LIGNE~
:";
52350 PRINT PEEK(AA+2)+256*PEEK(AA+3)
52360 PRINT"{RVS}";RIGHT$("{3SPACES}"+STR~
$(AA+3),5);"{OFF}="
52370 AA=AA+3
52380 REM ----- TEXTE
52390 AA=AA+1:PRINT"{RVS}";RIGHT$("{3SPAC~
ES}"+STR$(AA),5);"{OFF} "
52400 IF PEEK(AA)=0 THEN PRINT"{2SPACES}0~
", "FIN{SH.SP}DE{SH.SP}LIGNE":PRINT~
T$:AA=AA+1:GOTO 52290
52410 IF PEEK(KE)<>Z THEN GOSUB 53500
52420 PRINT RIGHT$(" "+STR$(PEEK(AA)),3),
52430 IF Q AND PEEK(AA)=34 THEN Q=0:GOTO ~
52460
52440 IF PEEK(AA)=34 THEN Q=1
52450 IF Q THEN PRINT C1$(PEEK(AA)):GOTO ~
52390
52460 PRINT C$(PEEK(AA)):GOTO 52390
52470 END
52480 REM =====
52490 REM SOUS-PROGRAMES
52500 REM =====

```

```

52510 REM.
52520 REM -----
52530 REM DEMO LIENS
52540 REM -----
52550 REM.
52560 GOSUB 52940:IF AA<>LM THEN L=1:C=6:~
N=3:GOSUB 53180
52570 PRINT"LES DEUX PREMIERS OCTETS D'UN~
E LIGNE DE"
52580 PRINT"BASIC CONTIENNENT L'ADRESSE D~
E DEBUT DE"
52590 PRINT"LA LIGNE SUIVANTE, DANS L'ORD~
RE"
52600 PRINT"POIDS FAIBLE / POIDS FORT:":G~
OSUB 53360:GOSUB 53360
52610 GOSUB 53360:FOR A0=0 TO 1:GOSUB 532~
90:NEXT A0:GOSUB 53360
52620 PRINT"L'ADRESSE ";RIGHT$("{2SPACES}~
"+STR$(AA),4);"($";:D=AA:GOSUB 530~
80
52630 PRINT") CONTIENT ";RIGHT$(" "+STR$(~
PEEK(AA)),3);"($";
D=PEEK(AA):GOSUB 53100:PRINT")"
52650 GOSUB 53360:C=13:GOSUB 53180:GOSUB ~
53360
52660 PRINT"L'ADRESSE ";RIGHT$("{2SPACES}~
"+STR$(AA+1),4);"($";:D=AA+1:GOSUB~
53080
52670 PRINT") CONTIENT ";RIGHT$(" "+STR$(~
PEEK(AA+1)),3);"($";
D=PEEK(AA+1):GOSUB 53100:PRINT")"
52690 GOSUB 53360:C=21:GOSUB 53180:GOSUB ~
53360
52700 PRINT"SOIT, DANS L'ORDRE NORMAL, $"~
;
52710 PA=PEEK(AA)+256*PEEK(AA+1):D=PA:GOS~
UB 53080:PRINT".":GOSUB 53360
52720 PRINT"$";:D=PA:GOSUB 53080
52730 PRINT"=-256*{RVS}";MID$(STR$(PEEK(AA~
+1)),2);
52740 GOSUB 53360:L=18:C=31:N=2:GOSUB 531~
80:GOSUB 53360
52750 PRINT"{OFF}+{RVS}";MID$(STR$(PEEK(A~
A)),2);
52760 GOSUB 53360:L=17:C=31:N=2:GOSUB 531~
80:GOSUB 53360
52770 PRINT"{OFF}=";MID$(STR$(PA),2):GOSU~
B 53360
52780 PRINT"LA LIGNE SUIVANTE COMMENCE EN~
";MID$(STR$(PA),2);"."
52790 GOSUB 53360:A0=PA-AA:GOSUB 53290:GO~
SUB 53420
52800 PRINT CE$;:FOR I=1 TO 11:PRINT B$:N~
EXT:PRINT CE$;
52810 PRINT"LA FIN D'UNE LIGNE DE BASIC E~
ST SIGNALEE";
52820 PRINT CR$;"PAR UN OCTET DE VALEUR 0~
.":GOSUB 53360
52830 GOSUB 53360:A0=PA-AA-1:GOSUB 53290
52840 PRINT"L'OCTET ";MID$(STR$(PA),2);",~
DEBUT DE LIGNE";
52850 GOSUB 53360:A0= PA-AA:GOSUB 53290
52860 PRINT" EST DONC":PRINT"PRECEDE D'UN~
OCTET CONTENANT 0."
52870 GOSUB 53360:A0= PA-AA-1:GOSUB 53290~
52880 RETURN
52890 REM.
52900 REM -----
52910 REM TABLEAU D'OCTETS
52920 REM -----
52930 REM.
52940 PRINT"{CLR}{RVS} ADRESSES ";TAB(13)~
; "{RVS}{8SPACES}CONTENU{8SPACES}"
52950 FOR I=AA TO AA+32 STEP 3
52960 PRINT"$";:D=I:GOSUB 53080:PRINT"=";~
RIGHT$("{3SPACES}"+STR$(I),4);"{2S~
PACES}";
52970 FOR K=0 TO 2
52980 PRINT" $";:D=PEEK(I+K):GOSUB 53100
52990 PRINT"=";RIGHT$(" "+STR$(PEEK(I+K))~
,3);

```

```

53000 NEXT K:PRINT:NEXT I:PRINT T$
53010 RETURN
53020 REM.
53030 REM -----
53040 REM DECIMAL>HEXA
53050 REM -----
53060 REM.
53070 REM SUR 2 OCTETS
53080 D%=D/256:D0=D-256*D%D=D%D:GOSUB5310~
0:D=D0:GOSUB53100:RETURN
53090 REM SUR 1 OCTET.
53100 D=D/16:FORJ=1TO2:D%=D:H$=CHR$(48+D%~
-(D%>9)*7):PRINTH$;:D=16*(D-D%):NE~
XT:RETURN
53110 REM.
53120 REM -----
53130 REM CLIGNOTEMENT
53140 REM DE N CASES-ECRAN, A PARTIR
53150 REM DE LA LIGNE L, COLONNE C
53160 REM -----
53170 REM.
53180 FOR KK=1 TO 5
53190 E=L*LE+C:FOR II=E TO E+N:POKE DE+II~
,PEEK(DE+II)-128*(PEEK(DE+II)<128)~
53200 IF MC THEN POKE MC+II,1
53210 NEXT II
    
```

```

53220 E=DE+L*LE+C:FORII=ETOE+N:POKEII,PEE~
K(II)+128*(PEEK(II)>128):NEXTII
53230 NEXT KK:RETURN
53240 REM.
53250 REM -----
53260 REM L'OCTET A0 CLIGNOTE
53270 REM -----
53280 REM.
53290 IF A0>34 THEN RETURN
53300 L=1+INT(A0/3):C=13+8*(A0-3*INT(A0/3~
)):N=6:GOSUB 53180:RETURN
53310 REM.
53320 REM -----
53330 REM TEMPORISATION
53340 REM -----
53350 REM.
53360 FOR TT=1 TO T:NEXT:RETURN
53370 REM.
53380 REM -----
53390 REM TOURNE-PAGE
53400 REM -----
53410 REM.
53420 PRINT CD$,,"{RVS}RETURN"
53430 GET R$:IF R$<>CHR$(13) THEN 53430
53440 :RETURN
53450 REM.
53460 REM -----
53470 REM PAUSE
53480 REM -----
53490 REM.
53500 GET R$:IF R$<>"THEN RETURN
53510 GOTO 53500
53520 REM.
53530 REM -----
53540 REM TOKENS
53550 REM -----
53560 REM.
53570 DATA END, FOR
53580 DATA NEXT, DATA, INPUT#, INPUT, DIM, REA~
D, LET, GOTO, RUN, IF
53590 DATA RESTORE, GOSUB, RETURN, REM, STOP, ~
ON, WAIT, LOAD, SAVE, VERIFY
53600 DATA DEF, POKE, PRINT#, PRINT, CONT, LIS~
T, CLR, CMD, SYS, OPEN
53610 DATA CLOSE, GET, NEW, TAB(, TO, FN, SPC(, ~
THEN, NOT, STEP
53620 DATA +, -, *, /, ^, AND, OR, >, =, <
53630 DATA SGN, INT, ABS, USR, FRE, POS, SQR, RN~
D, LOG, EXP
53640 DATA COS, SIN, TAN, ATN, PEEK, LEN, STR$, ~
VAL, ASC, CHR$
53650 DATA LEFT$, RIGHT$, MID$, GO, CONCAT, DO~
PEN, DCLOSE, RECORD, HEADER, COLLECT
53660 DATA BACKUP, COPY, APPEND, DSAVE, DLOAD~
, CATALOG, RENAME, SCRATCH, DIRECTORY
53670 REM
53680 REM -----
53690 REM CHR$ SPECIAUX
53700 REM -----
53710 REM.
53720 DATA 5, BLANC, 7, SONNERIE, 9, TAB, 13, RE~
TURN, 14, "MODE TEXTE"
53730 DATA 13, "EFF. APRES CRSR", 17, "CRSR ~
BAS", 18, RVS, 19, HOME, 20, DELETE
53740 DATA 20, "CRSR DROIT", 28, ROUGE, 30, VE~
RT, 31, BLEU, 32, ESPACE, 129, ORANGE
53750 DATA 131, RUN, 135, "DOUBLE SONNERIE"
53760 DATA 137, "TAB +/-", 141, "SHIFT RETUR~
N", 142, "MODE GRAPHIQUE", 144, NOIR
53770 DATA 145, "CRSR HAUT", 146, "RVS OFF", ~
147, "EFF. ECRAN", 148, "INSERT"
53780 DATA 149, MARRON, 150, "EFF.->CRSR (CB~
M){DOWN}{16LEFT}OU ROSE{4SPACES}{C~
64}"
53790 DATA 151, "GRIS FONCE", 152, "GRIS MOY~
EN", 153, "VERT PALE", 154, "BLEU PALE"
53800 DATA 155, "GRIS CLAIR", 156, VIOLET, 15~
7, "CRSR GAUCHE", 158, JAUNE, 159, CYAN~
53810 DATA -1
    
```

## CHANGEZ-MOI ÇA !

Nous sommes un certain nombre, je crois, à ne pas pouvoir copier un programme dans *L'Ordinateur Individuel* (ou d'autres journaux) sans avoir envie de le modifier au passage, pour l'adapter à nos goûts, à nos manies ou à nos besoins. Voici quelques directions possibles à suivre ou à ne pas suivre.

- Court-circuiter la première partie explicative : ajouter 50275 GOTO 51680.
- Court-circuiter les deux parties explicatives : ajouter 50275 GOTO 52120.
- Arrêter l'exploration d'un programme avant la ligne 50000 : ajouter 52295 IF PEEK (AA+2)+256\*PEEK (AA+3)=50000 THEN 52470.
- Se montrer nettement plus fûté que l'auteur : ajouter 50275 DEF FNP(X)=PEEK (X)+256\*PEEK (X+1) et remplacer toutes les lectures de valeurs sur deux octets par un simple FNP(X).
- Afficher les valeurs hexadécimales pendant l'analyse du programme : a vous de voir...
- Adapter le programme pour le VIC.
- Remplacer les lignes de REM du début par une adaptation automatique à la machine, grâce à quelques PEEKs discriminants judicieusement choisis.
- Sur tous les appareils, excepté le CBM 8000, l'écran est en mode majuscules/graphiques à la mise sous-tension. Comme le nom des machines est en caractères shiftés, il risque de ne pas sauter aux yeux. Alors, nous allons vous donner un truc, mais ne l'ébruitez pas.

Ajoutez une ligne 49999 REM ainsi composée. Frappez 499999 REM, puis un guillemet. Frappez la touche RETURN. Ramenez le curseur après le guillemet, et appuyez sur la touche RVS (CONTROL 9 sur le 64). Frappez ensuite SHIFT M et SHIFT N, puis la touche RETURN. C'est tout. L'affichage passera automatiquement en minuscules lorsque vous listerez votre programme. Ne demandez pas comment ça marche, c'est un truc de Weaselface et Weaselface est mort.



(Suite de la page 31)

virgule flottante. J'ai donc intérêt à écrire  $D0 \% = D/B$ . Le reste de l'opération, c'est tout bêtement  $D - (D0 \% * 256)$ , mais vu l'ordre dans lequel l'ordinateur fait les opérations (d'abord les exponentielles, puis les multiplications et les divisions, ensuite seulement les multiplications et les soustractions), je n'ai même pas besoin de parenthèses. Le résultat s'appellera  $D1 \%$ .

Dire qu'à l'école, j'étais nul en maths ! Quand on me demandait « trois fois douze ? », je répondais quarante parce que Joe Brown me soufflait quarante. Je prenais une sale note, je flanquais une baffé à Joe Brown, et Joe Brown pleurait en disant que c'était ce que son papa lui apprenait. Ce n'est que plus tard que j'ai appris qu'il était barman, le papa de Joe Brown...

**Lorsque deux nullités  
se rassemblent à la place  
d'un linéaire numéral,  
réjouissez-vous**

Si donc le numéro de la ligne de départ,  $D$ , est démonté en deux moitiés,  $D0 \%$ , l'octet de poids fort et,  $D1 \%$ , l'octet de poids faible, on va imposer par la menace ce contenu aux adresses qui contiennent le numéro de ligne, déjà connues sous le sobriquet de  $A+2$  et  $A+3$ . C'est tout pour l'adresse de départ. Le numéro de la ligne d'après, c'est le numéro de la ligne d'avant plus la différence entre la ligne d'avant et la ligne d'après, suis-je clair ? En un mot comme en cent,  $D = D+P$  et puis n'insistez pas, on n'a pas que ça à faire. A la suivante de ces lignes s'il vous plaît... Par ici... La suivante de ces lignes, j'avais déjà noté son adresse, je la jette et mets à sa place la nouvelle adresse :  $A = NA$ .

Au fait, qu'est-ce qui va me dire quand le programme se termine ? On lit dans L'Épître aux Commodoriens, chapitre 16, verset 256 : « Lorsque vous verrez se rassembler deux nullités en lieu et place d'un linéaire numéral, alors, réjouissez-vous, la fin du programme est proche. » Et il n'y est pas question de la télévision. C'est vrai, ça, que si le zéro qui marque la fin d'une ligne est lui-même suivi de deux zéros, cela signale la fin du Basic. Seulement, si on pense qu'à cet endroit-là, si le programme ne s'arrêtait pas, il y aurait l'adresse de la ligne suivante, on pourrait dire, comme ça, pour jouer, même que ça compterait pour du beurre, qu'un programme est fini quand l'adresse de la ligne, suivante est zéro. Alors, testons en chœur : `IF A = 0 THEN... Then quoi ? STOPper ?` C'est possible, mais comme d'une part mon premier geste sera de contempler l'aspect impeccable du programme renuméroté, et, comme d'autre part, chez Commodore, l'exécution d'un programme Basic ne reprend pas après une instruction `LIST`, autant mettre là un `THEN LIST`, boucler la boucle par un petit `NEXT` des familles, et on n'en parle plus !

Je lance la renumérotation, ma ligne pirate se fond dans le paysage. Parfait... Il y a là une boîte de disquettes. J'en remets une dans le drive, jette un coup d'œil à mon travail... Avec un peu de chance, ils ne s'apercevront de rien avant de lancer le programme. Et alors, il sera trop tard. Je referme la porte et sors du *Soliqueen's Club* par la porte de derrière. Il est temps que je rentre. Fondu au noir sur une silhouette, au trench-coat un peu informe, il est

## PROGRAMME DE RENUMEROTATION

Le programme de renumérotation hâtivement mis au point par Philip Microwe pour dissimuler le piège tendu à Tony Jennett. Il vous servira naturellement à des fins privées et non polémiques. Attention ! ce programme ne renumérote pas les lignes-cibles des `GOTO` et `GOSUB` (avec trois lignes, vous voulez rire !)

L'adresse de départ, la valeur de  $A$  à la ligne 63000, est supérieure d'une unité à celle donnée par le tableau de l'encadré « Quelques adresses utiles » p. 31. Pour un PET/CBM, on mettra  $A=1025$ , etc. D'accord, il n'est pas long à frapper après un programme existant, mais, même si vous ne disposez pas d'un `APPEND`, il est facile de l'ajouter en se servant de la mémoire d'écran : chargez le programme de renumérotation et faites `LIST`. Chargez ensuite le programme à renuméroter, et ne le listez surtout pas !

Le programme de renumérotation n'est plus en mémoire, mais il est encore sur l'écran. Remontez le curseur sur la ligne 63000 et frappez trois fois `RETURN`. La renumérotation est maintenant en mémoire. Faites `RUN 63000`, donnez le numéro que vous voulez imposer à la première ligne du programme renuméroté, frappez une virgule, puis le pas (10 pour numéroté de 10 en 10, par exemple), puis `RETURN`. Attendez un tout petit chouïa. Le programme modifié se listera tout seul, et vous n'aurez plus qu'à retoucher les `GOTO` et `GOSUB`.

vrai, mais non dépourvue d'une certaine élégance, s'éloignant dans l'obscurité complice de l'impasse, cependant que retentit encore le thème rauque du sax entendu précédemment.



Enfoncé dans mon trench-coat, je quittais le *Soliqueen's Club*. Le programme de renumérotation m'avait permis de glisser ma ligne pirate. La plainte du sax n'était qu'un prélude...

Mimi Flytox avait le chic pour trouver des rouges à lèvres à tartiner. Ce jour-là, c'était le genre bigarreau

## L'INTUITION DE MIMI FLYTOX

Mimi Flytox est une fichue menteuse ! L'emballage du sucre pour son café comportait un de ces « Le saviez-vous ? » qui sont la gloire de la culture occidentale, et où elle a appris ceci :

O REM	(RETURN)
POKE 1026,8	(RETURN)
0	(RETURN)
POKE 43, PEEK (43)	(RETURN)
CLR	(RETURN)

Sur PET/CBM, le Basic commence invariablement à 1024. Tout programme se charge à l'adresse où il a été sauvegardé.

Sur C.64, le Basic commence en 2048. Un programme sauvegardé chargé normalement est relogé en 2048, même s'il a été relogé à une autre adresse.

Sur VIC 20, le Basic commence à une adresse qui dépend de la configuration mémoire (cf. encadré « Quelques adresses utiles »). Un programme sauvegardé et chargé normalement est relogé au début de Basic, même s'il a été sauvegardé à une autre adresse. Moralité : Un VIC 20, ou un C.64 relogent automatiquement les programmes sauvegardés sur un PET/CBM, alors que l'inverse n'est pas vrai.

Donc, lorsque Philip Microwe a chargé sur son CBM le programme sauvegardé sur C.64, ce programme s'est installé à partir de 2 048 (\$0800), et comme le CBM le cherchait en 1024 (\$0400), il ne trouvait rien, le niais. Mimi Flytox a utilisé une ruse de guerre. Cette ligne-bidon se trouvait au début de Basic, en 1024, et était stockée ainsi :

en 1024, un zéro  
 en 1025-1026, l'adresse de la ligne suivante  
 en 1027-1028, le numéro de la ligne, c'est-à-dire zéro deux fois  
 en 1029, 143, c'est-à-dire REM  
 en 1030, un zéro marquant la fin de la ligne  
 en 1031-1032, deux zéros encore, puisque c'est la fin du programme.

L'adresse de la ligne suivante contenue en 1025-1026 était donc 1031, soit, en 1026, INT (1031/256) c'est-à-dire 4, et en 1025, 1031-(4\*256) c'est-à-dire 1031-1024, c'est-à-dire 7. Vous qui possédez maintenant un Basic-liste en état de marche (vous n'avez plus aucune excuse), ajoutez-y la ligne REM et faites tourner pour vérifier.

Mais, en frappant sa ligne, Mimi a fait reculer en mémoire le programme de Weaselface, qui s'est retrouvé en 2055. Quand elle a frappé POKE 1026, 8 en remplaçant par un 8 le 4 contenu en 1026, elle a fait croire à la machine que la ligne suivante allait commencer non pas en  $4*256 + 7 = 1031$ , mais en  $8*256 + 7 = 2055$ . Là-dessus, elle a supprimé sa ligne zéro. En moins de temps qu'il n'en faut à un bégue pour le dire, l'ordinateur a donc fait redescendre en 1025 ce qui était en 1055 : le Basic-liste de Weaselface. Seulement, l'ordinateur croit toujours que le programme se termine 1024 octets plus haut qu'en réalité. Sa dernière précaution a eu pour effet de rectifier les pointeurs et de tout remettre en place.

Pour charger sur CBM un programme chargé sur VIC 20 de base, il aurait fallu faire POKE 1026, 16, et ôter 12 du contenu de la case 43, pour un VIC 20 + 8/16K, POKE 1026, 18 et soustraire 14 du contenu de la case 43. Le VIC20 + 3K ayant son début de Basic à la même adresse que le CBM, aucune manipulation n'est nécessaire.

cœur-de-pigeon sous vernis marine six couches. Elle tapait à la machine sur une *Excalibur* du temps de Théodore Roosevelt et fut surprise de me voir à mon bureau si tôt le matin.

– On vous a volé votre montre, patron ? s'enquit-elle.

– Mimi, mon chou, je me demande toujours comment vous faites pour manger du pop-corn avec votre rouge à lèvres sans qu'il adhère au passage. Mais aidez-moi plutôt à déballer ça.

– Une gazinière ! Vous comptez vous reconverter dans la pâtisserie fine ?

– Miss Flytox, sachez d'abord que ceci s'appelle un CBM 4000 et mérite le respect. Ensuite, cet appareil n'est pas un jouet, il a un vrai clavier, pas un anticonfiote, par conséquent, le café lui est néfaste mais pas autant qu'à moi, merci vous êtes un ange.

Je pris le gobelet de carton qu'elle me tendit avec un sourire comme une balise d'aéroport, et me brûlai les amygdales avec bonheur.

– Et maintenant, au travail.

**Le LIST ne répondait pas, le RUN n'était guère plus causant**

Après avoir branché une série de câbles, je chargeai le Basic-liste de Weaselface avec un bon vieux SHIFT/RUN des familles. LOADING annonça l'écran, avant d'afficher READY.

– Regardez ça comme c'est beau, lancai-je, tout en frappant LIST.

– Fascinant... se repoudra Mimi Flytox.

L'écran n'avait pas bougé, sinon pour un READY de plus. J'allais m'exclamer qu'elle était ready, celle-là, mais un reste de pudeur me fit éviter d'être à ce point mauvais devant ma secrétaire. Le programme était pourtant en mémoire (même une 1541 n'aurait pas mis une minute trente pour ne rien charger), mais le LIST ne répondait pas et le RUN n'était guère plus causant.

– Vous avez une explication, belle enfant ?

– Vous allez m'en vouloir si je réussis mieux que vous, laissa-t-elle tomber avant de s'asseoir devant le clavier.

Elle frappa REM, puis la touche RETURN, POKE 1026,8 puis encore RETURN, 0 suivi de RETURN, et enfin LIST puis RETURN. Le programme défila à l'écran.

– Miss Flytox, comment avez-vous trouvé ça ?

Elle rougit.

– Je ne sais pas, une intuition... dit-elle.

Mais elle ajouta, comme par acquis de conscience : POKE 43, PEEK (43)– 4: CLR et entreprit de se repeigner.

Grattant une allumette sur la peinture écaillée de la fenêtre pour essayer de dissimuler au mieux ma stupéfaction, j'allumai avec beaucoup de peine un crayon HB qui traînait à côté de mes cigarettes, et regrettai, mais trop tard, qu'il comportât une gomme.

J'essayai de me calmer et de rassembler mes idées sur cette affaire lorsque l'une d'elles sortit du troupeau et vint se poser sur mon nez. Comment ne l'avais-je pas remarqué avant ? Lorsque Weaselface m'avait montré la version imprimée de son programme, les remarques, et elles seules, étaient en caractères expansés !

Comment avait-il pu s'y prendre ?



# VIDÉO RENVERSEE POUR ASSEMBLEUR SIMULÉ

**A propos de mots clés, appelés Tokens, on découvre un premier programme mutant. Un second fait apparaître les REMs en vidéo inverse.**

L'été est chaud, à Chicago. Il se trouve toujours un journaliste pour se croire original de publier la photo d'un gosse occupé à faire cuire un œuf simplement en le cassant sur une bordure de trottoir brûlante. L'eau glacée coule à flots dans les gobelets de carton des bureaux tandis qu'au plafond, d'énormes ventilateurs essaient, avec un bonheur inégal, de brasser un air, épais et gluant comme des nouilles trop cuites, qui s'agglutine aux pales et s'étire visqueusement.

Mais nous sommes à New York et par un frisquet mois d'avril. Assis devant un vieux CBM 4000, je cherche à comprendre le programme Basic-liste et en particulier comment Weaselface est arrivé à sortir un listing avec les REMs en caractères double largeur. Ça ne vient pas. Le vent souffle en rafales à travers les fenêtres disjointes du bureau décrépi. Mimi Flytox, ma pulpeuse secrétaire a des hanches à vous coller quarante de fièvre à un nourrisson aveugle de naissance, mais ça ne l'empêche pas d'avoir les synapses engourdies par le froid. Tiens, si seulement j'avais un Commodore 64 ! L'alimentation secteur suffirait à chauffer la pièce après un quart d'heure d'utilisation...

– Salut, Pat'on, ça vous intéresse ?

C'est Sigisbal N'bamguégué, le concierge de l'immeuble, qui vient d'entrer avec désinvolture, et un tas de cageots vides. C'est vraiment un type adorable. Il est allé faire le streetmarket de Maubert Square après la fermeture, et il en rapporte de quoi allumer le poêle.

– Oh, m'sieu Ci-gît, vous êtes un amour !

Précisons que c'est Mimi Flytox qui cause. Elle se jette au cou du bienfaiteur et l'embrasse sur les deux joues.

Après dix minutes que je passe à ranimer Sigisbal, le poêle ronfle, le moral revient, Sigisbal se barre et un doute m'étreint :

– Dites donc, Cendrillon, ce feu, vous l'avez allumé avec quoi, au juste ?

– Oh, de vieux papiers qui traînaient sur le bureau.

– Hell ! Le listing !

## Des éléphants verts phosphoreux dansaient sur l'écran

Dix minutes de plus pour convaincre Miss Flytox que se jeter par la fenêtre ne servirait qu'à faire rentrer le froid dans la pièce, deux rasades de bourbon pour m'en persuader aussi, me dire que je n'ai pas d'imprimante, mais que j'ai toujours le programme et la journée devant moi, que la mort ne vaut pas d'être vécue et que je serai un homme, mon fils.

C'est cette histoire de lignes de REMarques qui m'irrite un tantinet. D'autant plus qu'elles sont là pour m'aider à comprendre le programme, mais que je les mélange au texte et que si ça continue, je vais finir par voir de petits éléphants d'un vert phosphoreux danser sur l'écran.



D.B.

Je sais ! Je vais me les mettre en vidéo inversée. Comme ça, au moins, je ferai la différence. Allez RVS, je me refrappe la ligne, là au moins, c'est pétant, un petit RETURN des familles, un coup de LIST pour voir...

– Mimi, pensez-vous qu'une citerne de café me remettrait les idées en place ? Regardez ce qui m'arrive.

– C'est l'éditeur d'écran qui n'accepte pas les caractères inversés et leur rend leur valeur normale, il n'y a pas à s'étonner, répondit-elle au milieu d'un nuage de poudre de riz.

– Miss Flytox, savez-vous que quatre-vingt quinze pour cent des secrétaires au chômage sont celles qui ont été assez bêtes pour se montrer plus intelligentes que leur employeur ?

– Ne me faites pas pleurer, boss, j'ai un eyeliner neuf, geignit-elle, c'est seulement mon neveu qui m'a dit ça hier... Les remarques inversées, ça se peut pas, qu'il m'a dit. Je ne sais pas pourquoi ça m'est resté en tête. Elle se moucha énergiquement, puis enchaîna :

– Il a même dit qu'avec un SYS 4, on se sortait d'un mauvais pas sur deux. Mais je n'ai pas compris ce qu'il voulait dire.

SYS 4... le moniteur langage machine incorporé aux vieux CBM et qui fait si cruellement défaut aux

Miss Flytox avait oublié que les secrétaires ou chômage sont généralement celles qui se sont montrées plus intelligentes que leur patron.

**ENCORE DES TOKENS !**

Basic-liste donne une liste partielle des TOKENS, ces codes supérieurs à 128 qui permettent de stocker sur un seul octet chacun des mots clés de Basic. Un mini-programme permet de les avoir toujours sous la main et d'en sortir la liste sur imprimante à volonté.

128 REM

129 REM

130 REM

... etc. jusqu'à :

203 REM

500 A=2048+1:FOR I=128 TO 203: POKE

A+4,I:A=A+6:NEXT:LIST- 203

600 A=2048+1:FOR I=128 TO 203: POKE

A+4,143:A=A+6:NEXT:LIST - 203.

Ce programme commence par une floppée de lignes de REM numérotées de 128 à 203 (jusqu'à 218 si l'on dispose du Basic 4.0). Afin de s'économiser un peu, voici un raccourci qui met en œuvre les joies de l'éditeur plein écran des Commodore :

Frapper d'abord : 128 REM et (RETURN) puis un curseur-haut et deux curseurs-droite : vous êtes sur le '8'.

Frapper : 9 et (RETURN).

Frapper un curseur-haut et : 130 (RETURN).

Puis, à coups de « un curseur-haut, deux curseurs-droite », vous pouvez aller, sur la même ligne, jusqu'à 139.

Faire un LIST et amener le curseur en face de 130 et frapper dix fois : 14 (ou 014 sur un CBM).

On peut alors afficher 20 lignes par 20 lignes et on a vite atteint le 203.

Reste alors la ligne de programme :

500 A=2048+1:FOR I=128 TO 203: POKE

A+4,I:A=A+6:NEXT:LIST - 203.

La valeur de A est celle de début de Basic sur un 64. La modifier sur une autre machine.

Que fait donc ce bout de programme ?

En 2048, on a un 0, puisque c'est le départ de Basic.

En 2049, on a 7 et en 2050, 8.

7 plus 256 fois 8 = 2055, c'est l'adresse du début de la prochaine ligne.

En 2051, on a 128 et en 2052, 0.

128 plus 256 fois 0 = 128, c'est le numéro de la ligne.

En 2053, on a 143. C'est le code de REM.

En 2054, on a 0. C'est la fin de la ligne.

En 2055, on a 13 et en 1056, 8.

$13+(256*8)=2061$ , c'est l'adresse du début de la prochaine ligne, mais 2055, c'est bien l'adresse annoncée précédemment.

En 2057, on a 129 et en 2058, 0, c'est le numéro de la deuxième ligne.

En 2059, on a 143, code de REM et en 2060, 0=fin de ligne.

Le programme part de A, c'est-à-dire 2049. En A+4, là où nous avons 143 pour REM, il met 128, code de END.

Il recalcule A en A+6, soit 2055, et se place au début de la ligne suivante. A+4 sera cette fois 2059, à nouveau l'adresse de REM, qu'il remplacera pour 129, code de FOR. Et ainsi de suite. Quand il a fini, il liste, et nous voyons un programme impossible à exécuter, mais dont les mots clés sont des reconstitutions de synthèse parfaite !

Vous voulez défaire ce que vous avez fait ?

Ajoutez :

600 A+2048+1:FOR I=128 TO 203: POKE  
A+4,143:A=A+6:NEXT:LIST - 203

et frappez RUN 600. Vous retrouvez votre liste originale.

Ce que vous pouvez être calé, tout de même !

jeunots... mouais... on peut peut-être essayer. Regardons le début du programme.

.: 0400 00 22 04 50 C3 8F 20 20

.: 0408 20 23 23 23 23 23 23

.: 0410 23 23 23 23 23 23 23

.: 0418 23 23 23 23 23 23 23

.: 0420 23 00 43 04 5A C3 8F 20

Voyons voir... en \$0400, 00, début de Basic, en \$0401-0402, 22 et 04, ça fait \$0422, c'est l'adresse de la ligne suivante (exact, j'ai un zéro en \$0421)... ensuite, 50 et C3... \$C350, ça fait 50000 en décimal, c'est le numéro de la première ligne, ça baigne... 8F, c'est 143, c'est le code pour REM. Et derrière, j'ai deux fois 20, c'est-à-dire 32 en décimal, des espaces. Et si on remplaçait avec le moniteur un espace par une inversion vidéo, un CHR\$ (32) par un CHR\$ (18), c'est-à-dire en hexa \$20 par \$12 ? Yaquà tenter. Correction, sortie du moniteur par X et RETURN, LIST et....

- Mimi, ça vous dirait de dîner au restaurant ce soir ? Vous êtes une fille épatante, et je vous augmenterais bien si grand-papa ne m'avait pas interdit avant de mourir tout ce qui ressemble à un bon mouvement. Je crois que vous venez de me donner la clé du problème.

- Quel problème ?

Dehors, un ivrogne beugle Tramiel Blues sans se soucier du crachin obstiné qui n'atteint pas ses amygdales.

- Vous aviez évoqué quelque chose comme un restaurant, patron ?

- Une seconde, j'y suis presque.

**Elle fit cliqueter  
sur le clavier  
des ongles de danseuse  
cambodgienne**

Ça fait quatre heures que j'y suis presque. Quatre heures que je fais la chasse aux 8F sur l'écran pour les faire suivre d'un 12 et inverser mes REMs.

- Ce n'est pas pour vous embêter, mais j'ai peur de n'être pas assez habillée pour un réveillon.

- Un réveillon ?

- Il est minuit moins le quart et j'espère que mon neveu ne m'attend pas à la sortie de l'école. Si vous essayiez de faire ça par programme, j'ai faim.

- Par programme ! Je voudrais vous y voir.

- C'est rigolo, ce petit carré qui clignote tout le temps sur votre écran, on peut le faire bouger ?

Elle fit cliqueter sur le clavier des ongles de danseuse cambodgienne en riant du bruit que cela produisait.

- Oh ! je crois que j'ai tout effacé, je suis bête !

Je bondis. Pas de blagues, eh... Ah ! non, c'était seulement l'écran. Il ne faut pas dire des choses pareilles. Je me sens le cœur aussi poussif que le percolateur du rital d'en bas. Elle m'a fait peur.

- Regardez Philip, R comme Roméo, U comme Uniforme et N comme November et toc !

Mais qu'est-ce qu'elle fait ? Il n'y a plus rien sur l'écran, ni message ni curseur. Planté ! Elle l'a planté. Il ne faudrait pas tout de même qu'elle croie que...

On jurerait que ma gorge a des ongles incarnés. Adrénaline et sueurs froides. C'est la dernière fois que je laisse cette môme approcher de la bécane. Qu'est-ce que je fais, moi, maintenant, j'implose ou...

- Vous avez vu comme c'est joli, tout ce vert ?

Damnède, comme on dit dans les Rom Pols ! Je tourne dingue ou quoi ? Sur l'écran défile sagement une liste dont toutes les REMs sont en vidéo inversée.



Le loufiat s'empara de la formule magique gravée dans la crème chantilly.

D.R.

– Mais qu'est-ce que vous avez fait ?  
– J'ai voulu faire un programme, et tout a raté.  
– Comment, tout a raté ? Mais vous venez de faire en trois minutes ce que je cherchais depuis quatre heures.

– Ah ? Alors je crois que je prendrai une côte de bœuf. Avec une sauce au piment. Vous aimez la sauce au piment ?

Le *Louis XVI* était un restaurant français de la 1793<sup>e</sup> Avenue, pas encore très connu, mais qui ne devait pas tarder à l'être. Pour faire plus *Louis XVI*, le décorateur avait meublé la salle en *Henri II made in Atlanta*. La viande y était de première bourre, et je savais aussi que le bourbon y était excellent, c'était la moindre des choses. Mais leurs glaces étaient époustouflantes. A la troisième coupe Dauphine (sept parfums, meringue, amandes grillées, caramel liquide, chocolat chaud, un Niagara de chantilly et une ombrelle en papier japonaise), Mimi Flytox cala.

– J'arrête, il n'y aurait plus de place pour la tarte.

– Baby, c'est avec un plaisir sans mélange que je vous ai vu prendre une légère collation sous mes yeux attendris, mais si vous voulez avoir encore un patron demain, vous allez m'expliquer ce que vous avez fait pour les REMs.

– Vous ne trouvez pas qu'il est un peu tard ?

– Vous m'expliquez ça tout de suite ou vous êtes privée de dessert.

Elle soupira, et du bout de sa petite cuillère, écrivit sur la chantilly. J'y lus ceci :

```
60000 A=1024+1:B=256
```

```
60010 PA=PEEK(A)+B*PEEK(A+1):IF PA=0 THEN LIST
```

```
60020 IF PEEK(A+4)=143 AND PEEK(A+5)=32 THEN POKE A+5,18
```

```
60030 A=PA:GOTO 60010
```

– C'est tout ?

– Oui, c'est tout, en m'en veuillez pas, quand je n'ai pas mes verres de contact, je ne vois rien de ce que j'écris, alors il a fallu que je condense.

– Mademoiselle a terminé ? demanda le loufiat. Et sans attendre la réponse, il s'empara de la coupe et disparut.

**Le menton du loufiat  
fit entendre le bruit  
d'un bâton de chocolat  
qu'on plie en deux**

Alors s'éveilla en moi ce sixième sens qui m'a toujours signalé le danger, même sous les apparences les plus anodines. Et puis, là-bas, près du bar, cette

## MISS FLYTOX ET SES INTUITIONS

Elle commence à m'énervier, celle-là, avec ses airs de ne pas y toucher ! Enfin, examinons quand même ses élucubrations.

```
60000 A=2048+1:B=256
```

```
60010 PA=PEEK(A)+B*PEEK(A+1):IF PA=0 THEN LIST
```

```
60020 IF PEEK(A+4)=143 AND PEEK(A+5)=32 THEN POKE A+5,18
```

```
60030 A=PA:GOTO 60010
```

Elle part ici de 2048 sur un Commodore 64 ou de 1024 sur un CBM, c'est-à-dire le bas de Basic (voir encadré « Quelques adresses utiles » p. 32).

2049 (A) et 2050 (A+1) contiennent l'adresse de la prochaine ligne de Basic. On la stocke dans la variable PA. Lorsque le programme est terminé, on trouve après le zéro qui marque la fin de la ligne deux autres zéros qui marquent la fin du programme. Tout se passe donc comme si la prochaine adresse était zéro. Si tel est le cas, on liste, et l'on sait que l'ordinateur ne reprend pas le programme après un LIST.

2050 et 2051 (A+2 et A+3) contiennent le numéro de la ligne : ici on s'en fiche.

2052 (A+4) contient soit un caractère ASCII (cas de LET omis dans l'affectation d'une variable) soit le code ou TOKEN d'un mot clé de Basic. C'est donc cette adresse qu'il faudra tester.

Si A+4 contient 143, la ligne commence par l'instruction REM. Dans ce cas, si le caractère suivant (A+5) est un espace, on peut sans danger le remplacer par 18, code de l'inversion vidéo.

La dernière ligne remplace l'adresse de la ligne en cours par l'adresse de la ligne suivante, et boucle.

On ne peut pas hélas maîtriser la fin d'inversion vidéo, car le code est 146, c'est-à-dire un code supérieur à 128, que l'interpréteur Basic comprendrait comme un mot clé, en l'occurrence WAIT.

Si je puis me permettre de critiquer les éclairs de génie de Mimi Flytox, on pourrait améliorer sa ligne 60020. Elle demande un double test avec l'instruction AND à chaque tour de boucle :

```
60020 IF PEEK(A+4)=143 AND PEEK(A+5)=32 THEN POKE A+5,18
```

Si on écrit à la place :

```
60020 IF PEEK(A+4)=143 THEN IF PEEK(A+5)=32 THEN POKE A+5,18
```

le second test ne sera effectué que si le premier est vérifié. La boucle n'exécute donc systématiquement qu'un seul test et tourne alors plus vite. Et toc ! Non mais, pour qui elle se prend celle-là ?

Il faut cependant dire au crédit de son génie qu'elle a, sans le savoir, découvert également le moyen utilisé par Weasel-face pour avoir sur le listing imprimé des REMs en caractères expansés. Les caractères double-largeur de l'imprimante sont déclenchés par un CHR\$(14) et arrêtés par un CHR\$(15) ou un retour-chariot (\*). Il suffit de remplacer à la ligne 60020 la valeur 18 par 14 pour avoir un programme dont les REMs sortiront sur imprimante en double largeur. Vous pouvez d'ailleurs surprendre vos copains en leur envoyant des programmes ainsi modifiés, car le CHR\$(1) n'apparaît absolument pas à l'écran !

Rappelons enfin que l'encadré « programme de rénumérotation » donnait une méthode simplette pour ajouter un sous-programme de quelques lignes à un programme en mémoire en utilisant la mémoire d'écran.

\* CHR\$(14) et CHR\$(15) uniquement sur les imprimantes CBM mais le retour-chariot n'annule pas les caractères expansés et le CHR\$(14) fait passer l'écran en minuscules.

silhouette qui s'en allait du pas nonchalant caractéristique des gens pressés qui ne veulent pas que ça se voie. Je bondis, et n'eus que le temps de voir démarquer en trombe une voiture rouge. C'était une Buick 12 litres 6 dernier modèle, équipée de pneus à flancs blancs... Quelque chose me disait que la coupe de Mimi n'était pas à la plonge, mais que l'inscription dans la chantilly ne manquerait pas d'amateurs.

— Pardon, Monsieur, mais c'est privé, par ici, c'est la cuisine.

Ai-je les phalanges plus dures en avril que le reste de l'année ? Toujours est-il que le menton du loufiat fit entendre le bruit d'un bâton de chocolat glacé qu'on plie en deux. Un éclair illumina soudain la cuisine.

C'était le flash d'un appareil photo et le modèle n'était autre que la coupe de chantilly. Le photographe ne prit pas le temps de doubler la prise et s'élança vers la sortie. Peu habitué à la côte de bœuf, je sprintai trop mou pour le suivre, mais j'eus quand même le temps de le voir sauter dans un taxi et d'apprendre où il allait. Pas vraiment une surprise. Sept secondes plus tard, je sautai à mon tour dans un cab.

— A l'*Ordinary Inquirer*.

En face de l'immeuble du Conservative Party, à Fabian Square, les bureaux de l'*Ordinary Inquirer* occupaient un immeuble neuf, après avoir abandonné le célèbre quartier de Beauties' Barn. Si mon intention première avait été de me précipiter dans les lieux tête baissée, quitte à friper quelque vieille dactylo au passage, un seul coup d'œil sur le hall m'aurait fait changer d'avis : une demi-douzaine d'appariteurs musclés se faisaient les ongles avec application, et, à en juger par la protubérance caractéristique qu'ils présentaient à la hauteur du holster d'aisselle, on avait dû leur faire copier vingt fois « Il ne faut pas plaisanter avec la liberté de la presse ».

**« Au poker, un bon joueur est un mauvais joueur »**

Tony Jennett était de mèche avec Bernie Soap, c'était certain. Dès qu'il avait compris que le travail avait été déjà fait par Weaselface, il avait décidé de ne plus payer mes services au tarif syndical, mais d'utiliser des méthodes plus directes. L'ennui, c'est que je ne vois pas le moyen de l'empêcher d'y parvenir, et que j'ai l'impression qu'avec les atouts qu'il a dans sa manche, je suis en train d'en perdre une. C'est une situation que je n'aime pas. « Au poker, disait grand-papa Microwe avec son bel accent traînant américano-vaudois, un bon joueur, c'est un mauvais joueur. » Ce jour-là, je compris enfin ce que ça voulait dire : quand on n'aime pas perdre, on n'a qu'à gagner. Par n'importe quel moyen. N'importe lequel. Les yeux rivés au sol d'un air buté, je faisais les cent pas devant l'immeuble, sous le regard vitreux d'un Irlandais visiblement saoul de naissance, bouteille de whisky demi-pleine à la main. Eh bien, savez-vous qu'on en apprend des choses, en fixant le trottoir ? J'ai appris, ce soir-là, qu'il y avait sept soupiraux à la façade de l'*Ordinary Inquirer*. Ce n'est pas le genre de connaissances qui permet de briller en société, mais sur les sept, l'un d'eux avait les vitres badigeonnées en noir. Drôle d'idée ! A moins que... Et si c'était le labo photo ?

Il y a des jours où des décisions rapides s'imposent. Un Irlandais + un soupirail = eurêka. Paddy, tu me



A l'*Ordinary Inquirer*, la liberté de la presse était solidement défendue ; heureusement, il y avait le whisky irlandais...

files ta bouteille, je sacrifie un mouchoir répugnant certes, mais sec, un coup de latte dans les vitres peintes du soupirail, un cocktail Irishov dans le trou : il y eut dans le labo suffisamment de lumière pour voiler tous les films qui s'y trouvaient, et même un peu de chaleur en plus. Ils n'avaient plus de trace des explications de Mimi Flytox.

Ce fut chez moi, sous la douche, les idées remises en place par l'eau glacée que l'idée me vint de faire preuve à l'égard de Tony Jennett d'un scrupule professionnel qu'il n'apprécierait peut-être pas.

April Fools Day approchait à grands pas. A la sueur de mon cortex, j'avais fini par maîtriser le programme Basic-liste que je connaissais maintenant dans ses moindres détails. Restait à en négocier la livraison à ce cher Tony Jennett. Les fesses en biais sur les ressorts fatigués de mon vieux fauteuil de bureau, je grattai une allumette en me promettant de m'offrir un fauteuil neuf dès que Jennett m'aurait payé. Miss Flytox, ma secrétaire, pourrait avoir le mien. D'un index ferme, je composai le numéro du *Soliqueen's Club*.

— *Soliqueen's Club* j'écoute...

La voix était maquillée à la limite inférieure de l'excès, et on devinait un parfum cher.

— Un ami voudrait parler à Tony.

— Je regrette, M. Jennett est absent en ce moment.

— Pinnochia, vous êtes incorrigible. Et ça s'entend quand vous mentez. Dites lui que c'est un ami. Un ami de Weaselface.

Il y eut au bout du fil un grésillement surpris.

— Je vous écoute, M. Microwe.

C'était bien la voix de mon type : de la flanelle rayée calibre 38.



## ÇA RESSEMBLE AU LANGAGE MACHINE

Ça a le goût du langage machine, mais ce n'est pas du langage machine. Pour charger le programme à protéger contre le listage, ajoutez la ligne suivante :

PET/CBM	OSYS1040
VIC de base	OSYS4112
VIC +3 K	OSYS1040
VIC +8 ou 16K	OSYS4264
C.64	OSYS2064

sans espace entre l'instruction SYS et le nombre.

Ajoutez ensuite la ligne :

1REM... (REM suivi de 11 points sans espace). Puis livrez-vous à la série de POKES correspondant à votre machine (voir Table 1). Il s'agit simplement d'un tout petit bout de programme en langage-machine de onze octets, une misère, qui suffit à tromper l'ennemi.

Faites LIST -1. Vous devez voir la ligne 0 suivie d'une ligne 1 ainsi formée :

1 REMSTEPGOSUBSTEPGOSUB

Si vous faites LIST tout court, la suite du programme apparaît. C'est le moment de faire les deux POKES de la Table 2. Vous pouvez enfin faire LIST : seul le SYS apparaît.

### Tableau 1

PET/CBM	VIC nu	VIC+3K	VIC+8/16K	C.64
POKE 1040,169	POKE 4112,169	POKE 1040,169	POKE 4624,169	POKE 2064,169
POKE 1041, 28	POKE 4113, 28	POKE 1041, 28	POKE 4625, 28	POKE 2065, 28
POKE 1042,141	POKE 4114,141	POKE 1042,141	POKE 4626,141	POKE 2066,141
POKE 1043, 11	POKE 4115, 11	POKE 1043, 11	POKE 4627, 11	POKE 2067, 11
POKE 1044, 4	POKE 4116, 18	POKE 1044, 4	POKE 4628, 16	POKE 2068, 8
POKE 1045,169	POKE 4117,169	POKE 1045,169	POKE 4629,169	POKE 2069,169
POKE 1046, 4	POKE 4118, 18	POKE 1046, 4	POKE 4630, 16	POKE 2070, 8
POKE 1047,141	POKE 4119,141	POKE 1047,141	POKE 4631,141	POKE 2071,141
POKE 1048, 12	POKE 4120, 12	POKE 1048, 12	POKE 4632, 12	POKE 2072, 12
POKE 1049, 4	POKE 4121, 18	POKE 1049, 4	POKE 4633, 16	POKE 2073, 8
POKE 1050, 96	POKE 4122, 96	POKE 1050, 96	POKE 4634, 96	POKE 2074, 96

### Tableau 2

PET/CBM	VIC nu	VIC+3K	VIC+8/16K	C.64
POKE 1035,0	POKE 4107,0	POKE 1035,0	POKE 4619,0	POKE 2055,0
POKE 1036,0	POKE 4108,0	POKE 1036,0	POKE 4620,0	POKE 2026,0

### Tableau 3

	PET	CBM	VIC	C.64
Inhiber la touche STOP	POKE 537 PEEK(537)+3	POKE 144 PEEK(144)+3	POKE 788 PEEK(788)+3	POKE 788 PEEK(788)+3
Inhiber STOP/RESTORE			POKE 37150,0	POKE 828,225
RESET	SYS 64824	3000 SYS 64721 4/8000 SYS 64790	SYS 64802	SYS 64738

Sur les PET/CBM, il suffit de frapper RETURN en réponse à un INPUT pour sortir du programme. Pour éviter cette éventualité, qui permettrait à vos victimes de s'apercevoir de la supercherie, ouvrez un canal pour le périphérique clavier :

OPEN 1,0

et faites vos INPUT sur ce canal :

INPUT # 1,AS

Attention aux effets secondaires : c'est à vous d'ajouter un point d'interrogation et un changement de ligne après la réponse si vous les souhaitez.

Bernie Soap eut ma peau et la disquette. Mais j'avais maquillé mon Basic en L.M.; j'aurais bien voulu voir sa tête!



D.B.

- Mission accomplie, boss. Je suis en mesure de vous livrer le programme.
- J'en suis... ravi, M. Microwe, je vous attends à mon bureau.
- C'est que je crains d'avoir dépassé mon budget... Weaselface s'était collé pour vingt ans de crédit sur le dos avec sa maison de Taybow Square et ce serait un beau geste de la part de Jennett que d'aider sa veuve si gentiment.
- Vous comprenez, boss, j'ai eu quelques frais de blanchisserie et...
- Je vous attends.
- Désolé, boss, mais je me suis laissé dire qu'il y a chez vous des gens qui fument. J'ai les bronches

### LA CLE DU MYSTERE

Le programme en langage-machine sur le C. 64 tient en quelques lignes :

LDA # \$1C  
STA \$080B  
LDA # \$08  
STA \$080C  
RTS

Que se passe-t-il ? Les deux POKEs de la table 2 arrivent à l'adresse où l'interpréteur Basic cherche l'adresse de la ligne qui suit la ligne 0. Ils y mettent des zéros, ce qui fait croire à ce grand benêt d'interpréteur Basic que le programme s'arrête à la ligne 0, et il ne liste pas plus avant. Quand on fait RUN, on lance le petit trognon de programme en langage machine qui ne fait rien d'autre que remettre en place l'adresse de la ligne suivante, ce qui rend au programme sa dimension normale. Tony Jennett n'y a vu que du feu.

fragiles, et je suis sûr que vous-même serez heureux de respirer l'air de la mer. Je serai à onze heures devant le dock de Look.

- Il n'en est pas question. D'ailleurs...

La voix n'avait rien perdu de son calme, mais elle était devenue moins nette pour moi depuis que j'avais déposé le téléphone dans le tiroir de gauche, plein de vieux rubans de machine à écrire.

Il me restait quatre heures pour réaliser une farce innocente : Jennett aurait son programme, même s'il devait le payer cher, et ce programme tournerait exactement comme prévu. Seulement, le cher homme ne pourrait ni le lister ni le comprendre : il ignorait tout du langage machine !

Moi aussi d'ailleurs, ou presque, et c'était bien là le plus drôle. Si j'étais passé la veille chez Bill, le neveu de Mimi Flytox, un rouquin à lunettes qui aurait fait avaler son képi au responsable des services informatiques du Pentagone, ce n'était pas totalement désintéressé. Car j'avais appris une ficelle genre fripon dont mon client aurait la primeur.

Un programme en langage machine, ça se lance depuis le Basic au moyen de l'instruction SYS. Quand on fait LIST, on a un numéro de ligne, et SYS suivi de l'adresse de début, c'est tout. C'est peu. Le Basic, c'est bavard, ça se met à table tout de suite. Mais le langage machine, c'est comme les Sysiliens : pour savoir ce que ça sait, un programme LM, il faut se lever de bonne heure, à moins de savoir causer le vernaculaire, ce qui n'est certainement pas le cas de mon client. Ce que j'avais appris chez ce boutonneux de Bill, c'était simplement l'art de maquiller un bon vieux programme Basic en programme LM. Pour son âge, il avait déjà un beau talent de faussaire, Bill, et le coup qu'il m'avait appris était à s'y méprendre. J'étais prêt à parier mon meilleur calibre contre un pistolet à flèches que Jennett ne s'apercevrait de rien. La première chose à faire est de rendre le programme étanche : désactiver le cran de sûreté de la touche STOP, avec ou sans RESTORE, et empêcher sur les CBM de sortir du programme sur un RETURN en réponse à INPUT. Naturellement, à la fin du programme, au lieu d'un STOP, d'un END ou d'un rien du tout, on pourra mettre un NEW, ou mieux encore, un RESET (voir encadré « ça ressemble à du langage machine... »). Là-dessus, le maquillage commence. Une ligne de SYS, une ligne de REM, treize POKEs, exactement, et voilà Karloff transformé en Frankenstein...

Dix minutes me suffirent pour modifier le programme et préparer une disquette. Je glissai un Remington dans ma ceinture, et bouclai mon trench.

Le dock de Look est vraiment un endroit sinistre. Murs de briques suintantes, pavés gras, squelettes décharnés des grues menaçantes, lugubre clapotis d'une eau huileuse où les rats ont du mal à se frayer un passage entre les gangsters noyés et les cadavres de privés. Sous un pont roulant, une élégante silhouette, aussi déplacée dans ce décor qu'une moissonneuse-batteuse dans un vol-au-vent un soir de dîner chic, c'était Tony Jennett.

- Bonsoir, Monsieur Jennett. Vous êtes seul ?
- Non.

Au même moment, une détonation retentit, une balle m'atteignit dans le dos, traversa ; je mourus.

Je ne vis pas - et pour cause - s'approcher une autre silhouette : celle de Bernie Soap. Ils n'eurent aucun mal à trouver sur moi la disquette. Mais la tête qu'ils vont faire en découvrant son contenu !

François-Jean Bayard



# LIST

## LE JOURNAL DES AMATEURS DE PROGRAMMATION

# LIST

## LE JOURNAL DES AMATEURS DE PROGRAMMATION

Dans  
chaque numéro :

- les "trucs" de votre machine.
- des programmes commentés.
- Basic, Forth, Pascal et les autres langages.
- des problèmes de programmation.
- les nouveautés : matériels, logiciels, programmes.



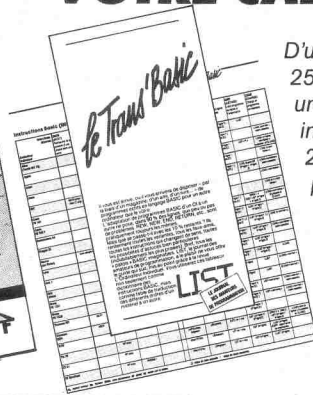
**Si** programmer un ordinateur est devenu pour vous un loisir, un plaisir...

une passion, sachez que **LIST** a été créé pour vous. **LIST** vous aide à tirer davantage de votre matériel, à vous perfectionner dans la conception des programmes qui "tourneront" sur votre machine.

**LIST** vous initie aux langages informatiques et sélectionne les meilleurs livres pour progresser. **LIST** vous informe de l'actualité et vous fournit trucs, astuces et idées pour mieux programmer...

Pour être sûr de ne rater aucun numéro et pour recevoir **LIST** chez vous, **abonnez-vous!**

### VOTRE CADEAU!



D'un format pratique (11,5 x 25 cm), le **TRANS'BASIC** est une table de conversion des instructions **BASIC** des 21 micro-ordinateurs les plus courants. Ce livret de 24 pages vous permettra d'adapter sur votre ordinateur la plupart des programmes conçus pour les autres machines.

# 40F D'ECONOMIE

Belgique: 160 FB - Canada: 2,95\$ - Suisse: 6 FS



O170

### BULLETIN D'ABONNEMENT

(à retourner à LIST - Service Abonnements - 5, place du Colonel Fabien - 75491 Paris Cedex 10)

Nom \_\_\_\_\_ Prénom \_\_\_\_\_

Adresse \_\_\_\_\_

Ville \_\_\_\_\_ Code postal | | | | | Pays \_\_\_\_\_

Veillez m'abonner pour 10 numéros au prix avantageux de 160 F\* au lieu de 200 F - prix total au numéro - et m'adresser en cadeau le **TRANS'BASIC**. Je joins mon règlement indispensable libellé à l'ordre de **LIST**.

\* Belgique: 1330 FB; Suisse: 50 FS; Canada: 30 \$C; autres pays: 210 FF. Par avion: Afrique Francophone: 245 FF; Amérique, autre Afrique, Océanie: 305 FF; Asie: 355 FF. Belgique: Soumillion, 28, av. Massenet, 1190 Bruxelles. Versement Société Générale 2100405 835-39. Suisse: 19, route du Grand-Mont, CH 1052, Le Mont-sur-Lausanne, versement Caisse d'Epargne et de Crédit, 10-2418 Le Mont CH 1052, compte courant n° 650156-7. Canada: LMPI, 9345, rue de Meaux, St Léonard (Québec), H1R 3H3, Canada. Autres pays: 5, place du Colonel Fabien, 75491 Paris Cedex 10.

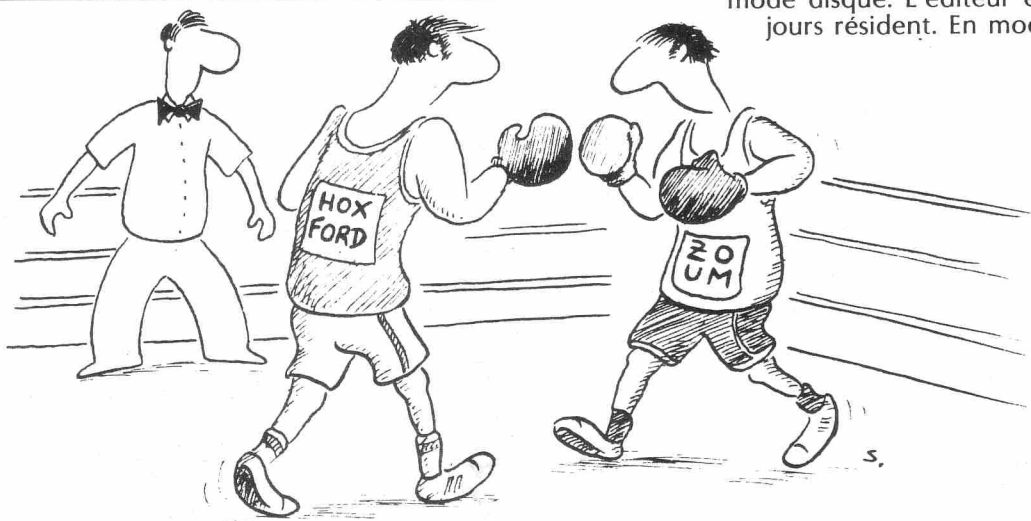
# PASCAL

## LE MATCH ZOOM-OXFORD

**Il traîne sur le marché deux compilateurs Pascal pour C. 64 : Zoom Pascal, importé par Micro-Application, et Oxford Pascal, disponible en Angleterre et, peut-être, en France.**

est traduit en langage machine (L.M.) du 6502. On peut l'exécuter par RUN, mais aussi le sauver par SAVE...

Quatrièmement : le programme en langage machine ainsi sauvé sera rechargé et exécuté à partir de Basic. Oxford Pascal fonctionne sous deux modes : le mode résident et le mode disque. L'éditeur est toujours résident. En mode rési-



**L**es deux produits se présentent sous la forme d'une disquette accompagnée d'une brochure d'Oxford. Celle de Pascal est plus luxueuse et plus détaillée « *but* » en anglais. Celle de Zoom Pascal est écrite dans la langue de Molière.

Les disquettes de l'un ou de l'autre produit sont protégées. Sur Oxford Pascal, la disquette ne supporte aucun « back-up. ». L'éditeur et le traducteur de Zoom Pascal sont copiables (par LOAD puis SAVE), mais pas le compilateur. Je suis personnellement contre cette forme de protection, car l'utilisateur a réellement besoin de copies de sécurité. J'aurais, et cela n'engage pas l'Ordinateur Individuel, préféré une protection par clé.

Zoom Pascal exige quatre étapes pour l'exécution d'un programme :

Premièrement : le chargement de l'éditeur. RUN. Création (ou modification) du programme source (éventuellement, chargement du programme à modifier). Stockage du programme source terminé.

Deuxièmement : le chargement du compilateur. RUN. Le programme source est compilé à partir du disque. Un fichier en P code (code intermédiaire) est écrit sur le disque. Pour faire une correction d'erreur à ce niveau, il faut éteindre et rallumer l'ordinateur et recharger l'éditeur !

Troisièmement : le chargement du traducteur, qui se fait sous contrôle du compilateur s'il n'y a pas eu d'erreur de compilation. Le programme

dent, le compilateur l'est aussi. Pour exécuter un programme, il suffit donc :

a) de charger Pascal. On entre ensuite le texte source. (On charge éventuellement le programme source à modifier).

b) de compiler par une commande « r » pour exécuter le programme.

En cas d'erreurs, le programme source est toujours correctible.

**Charger Pascal à votre mode et compiler le programme source**

Le mode résident est donc relativement semblable au fonctionnement de Basic. Pour des programmes plus longs qui utilisent toutes les ressources du langage, il faut être en mode disque. L'exécution nécessite trois étapes :

La première : édition du programme source, comme précédemment. Là, il faut obligatoirement ranger le texte sur disque.

La deuxième : compilation par la commande COMP.

La dernière : exécution par la commande EX.

On peut aussi compiler par LOCATE, ce qui crée un programme exécutable à partir de Basic. Mis à part le fait que le mode résident est bien pratique pour les débutants, qui doivent sans cesse modifier leurs programmes, je pense que l'éditeur d'Oxford Pascal est plus commode que



celui de Zoom Pascal. Les commandes sont plus simples. Elles ressemblent à celles de Basic (avec en plus la numération automatique, FIND, etc.). On peut même exécuter des instructions Basic en mode direct. Le mode d'emploi de l'éditeur Zoom de la brochure me semble très insuffisant.

En revanche, le programme traduit par Zoom Pascal est en vrai langage machine 6502 (on appelle cela du « code natif »), alors qu'Oxford Pascal interprète le P-code. Donc Zoom Pascal est *beaucoup plus rapide* (d'un facteur 10 à 50). Oxford Pascal n'est, à l'exécution, qu'un peu plus rapide (5 à 20 %) que Basic interprété.

### Touche pas au Zoom Pascal en enseignement

C'est sur ce point que se fait le véritable choix. Nous dirons qu'à ce niveau Zoom Pascal a du bon et du mauvais. Du côté du bon : quelques extensions de langage dont les principales touchent aux chaînes de caractère. On a le type STRING et les procédures chaînes du Pascal UCSD. Du côté du mauvais : il y a, par endroits, de sérieuses restrictions : absence d'ELSE (en « Pascal » !), de RECORD, de SET, de Pointeur, de FILE, pas de types structurés dans TYPE, ARRAY (à une dimension seulement).

Les restrictions rendent l'emploi de Zoom Pascal impossible en enseignement. Même en Basic, les tableaux sont en plusieurs dimensions ! RECORD est une caractéristique de Pascal !

Au niveau du langage, au contraire, Oxford Pascal nous a semblé parfait. Il s'agit d'un Pascal extrêmement complet, dont le point faible réside dans le traitement des chaînes de caractères. Ces réserves effectuées, toutes les possibilités de Pascal sont là : RECORD, TYPES, FILE, pointeurs, SET, etc. De fait, Oxford Pascal n'est autre que la version C. 64 du Pascal TRANSAM qui était un excellent Pascal sur CBM.

Oxford possède, de plus, un certain nombre d'extensions intéressantes : possibilité de manipuler l'interface IEEE série du C. 64, donc commandes disque (y compris l'accès direct) et imprimante. Commandes sonores et graphiques Haute Résolution.

Si je désirais travailler en Pascal, j'adopterais donc Oxford Pascal et non Zoom Pascal. Seul Oxford Pascal donne accès aux possibilités réelles du langage. Son mode résident est extrêmement pratique en contexte enseignement. Son éditeur de programmes sources est, lui aussi, beaucoup plus agréable d'emploi. Enfin, ses extensions graphiques concurrencent certaines cartouches d'extension Basic. Le seul intérêt de Zoom Pascal réside dans sa rapidité d'exécution (code natif).

En somme, vous pourriez plutôt l'utiliser comme compilateur Basic (à condition de mettre des ; en fins de ligne !). Et encore, son éditeur est désagréable. ■

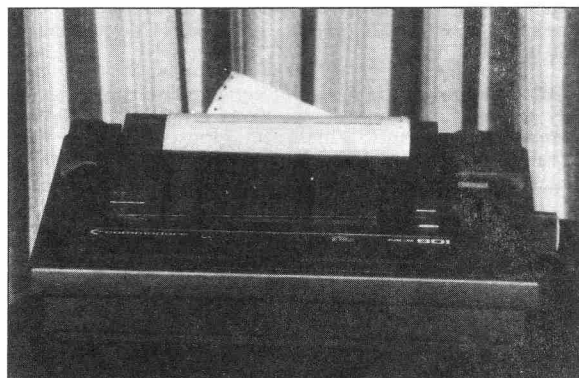
Pierre-Etienne Thalberg

NOTE. - Zoom Pascal : Micro-Application, 147, avenue Paul-Doumer, 92500 Rueil-Malmaison. Oxford Pascal : Oxford Computers, Hensington Road, Woodstock, Oxford, OX7 1S4, England.

# COPIE

## LA HR SUR IMPRIMANTE

**Impossible de recopier sur imprimante MPS 801 les écrans créés à partir de « Tool » ou « Koala Pad » ? Impossible de retranscrire en moins de trois minutes les dessins inventés avec « Simon's Basic » ? NON ! Notre programme pour tous logiciels graphiques tourne dans des délais record en Haute Résolution.**



MPS 801 : quelques acrobaties, mais des résultats en Haute Résolution.

Après avoir tapé et sauvegardé le programme, on fera « RUN ». Si les DATA ont été frappés correctement, l'écran indique les sommes de vérification suivantes :  
6683 6615 6701 6829  
6030 6753 9680

Dans le cas contraire, il ne reste qu'à chercher l'erreur, et à la corriger. On utilise la routine en commençant par préciser par un POKE l'adresse de l'écran Haute Résolution. L'encadré, indique les valeurs à « POKER » selon l'adresse de l'écran. Pour recopier un écran de « Koala Painter », on fera POKE 52613,96. La valeur par défaut de l'octet 52613 est 224, ce qui correspond aux écrans de « Tool » et de « Simon's Basic ». (Dans ce cas, le POKE est inutile). Si l'on désire recopier un écran de « Koala Painter », il faut le charger par LOAD « XXX », puis lancer le programme de copie par SYS52608. Pour recopier un écran dessiné avec « Simon's Basic » il suffit de faire SYS52608. Il est à noter que le programme qui a servi à implanter la routine peut s'effacer une fois la manœuvre effectuée.

Pour recopier un écran de « Tool », il est nécessaire de dessiner l'écran, puis de charger le programme de copie, de l'implanter (par RUN), et enfin de le lancer par SYS52608. Faute de procéder dans cet ordre, certaines instructions de « Tool » se modifient et deviennent inutilisables. ▶

Le fait d'être obligé d'enregistrer le programme de recopie et donc d'écraser le programme qui a permis de dessiner une image sur l'écran est fort gênant. On évite cet inconvénient avec un moniteur hexadécimal (par exemple « Zoom » ou « Mon » ; il suffit de sauvegarder la routine en langage machine implantée de CD80 à CE00 par une instruction du type :

S « NOM » 08 CD80 CE00

Cette routine sera récupérée sans écraser le programme de dessin par LOAD « NOM », 8, 1. La remarque vaut pour tous les autres logiciels dessin. ■

René Tisserand

Notes :

**TOOL 64** : diffusé, entre autres, par Procep (9, rue Sentou, 92150 Suresnes) et Micro-Application (147, rue Paul-Doumer, 92500, Reuil-Malmaison). Prix : 640 FF.

**KOALA PAINTER** : logiciel + tablette, importateur B.I.P. (13, rue Le-Duc, 75018 Paris). Prix : 1 300 FF.

**SIMON'S BASIC** : édité par Procep, distribué et vendu entre 200 et 750 FF chez tous les bons distributeurs.

**MON** : publié dans le « Livre du 64 », de Benoît Michel, édité chez PSI, (300p. 120 FF), 5, place du Colonel-Fabien, 75010 Paris.

**ZOOM** : Edité par « Supersoft », vendu par exemple dans les F.N.A.C. et chez Innelec. Prix : 196 FF en cassette, 235 FF en disquette. Supersoft Winchester House, Canning road, Wealdstone, Harrow HA3, 75G, England.

ADRESSE HEXA	ECRAN DECIM.	EXEMPLE	VALEUR A POKER en 52613
\$0000	0		0
\$0800	2048		8
\$1000	4096		16
\$1800	6144		24
\$2000	8192		32
\$2800	10240		40
\$3000	12288		48
\$3800	14336		56
\$4000	16384		64
\$4800	18432		72
\$5000	20480		80
\$5800	22528		88
\$6000	24576	KOALA-PAINTER	96
\$6800	26624		104
\$7000	28672		112
\$7800	30720		120
\$8000	32768		128
\$8800	34816		136
\$9000	36864		144
\$9800	38912		152
\$A000	40960		160
\$A800	43008		168
\$B000	45056		176
\$B800	47104		184
\$C000	49152		192
\$C800	51200		200
\$D000	53248		208
\$D800	55296		216
\$E000	57344	SIMON'S , TOOL	224
\$E800	59392		232
\$F000	61440		240
\$F800	63488		248

SAVE

```

1 ;SAVE"@SC0P52608".8
2 ;
3 ;
4 ECRANH=#E0
5 PSET=#FB
6 PSETH=#FC
7 PPSET=#FD
8 PPSETH=#FE
9 BISET=#FF
10 PDOT=#58
11 PDOTH=#59
12 CTROU=#5A
14 CCHAR=#5C
15 DRAPD=#5D
18 CPIX=#5E
19 ;
20 CHROUT=#FFD2
21 CLOSE=#FFC3
22 CLRCHN=#FFC0
23 CHKOUT=#FFC9
24 ;
25 ;
26 ;
27 ;
CD80 A900 28 LDA #00
CD82 85B7 29 STA #B7 ;LONGUEUR NOM DE FICHIER =0
CD84 85B9 30 STA #B9 ;ADRESSE SECONDAIRE
CD86 A904 31 LDA #04
CD88 85B8 32 STA #B8 ;NUMERO FICHIER LOGIQUE
CD8A 85BA 33 STA #BA ;NUM PERIPHERIQUE
CD8C 20C0FF 34 JSR #FFC0 ;OPEN4,4,0
35 ;
36 ;
37 ;
CD8F A204 38 LDX #04
CD9120C9FF 39 JSR CHKOUT ;UNITE 4 EN SORTIE
40 ;
41 ;
100 ;
102 ;
CD94 A900 106 LDA #0
CD96 85FB 108 STA PSET ;
CD98A9E0 110 LDA #ECRANH ;
CD9A 85FC 112 STA PSETH ;
CD9C A900 114 LDA #10000000 ;
CD9E85FF 116 STA BISET ;
CDA0 A91C 118 LDA #28
CDA2855C 120 STA CCHAR ;
CDA4 A900 122 LDA #0
CDA6855D 124 STA DRAPD ;
CDA8 A908 128 LDA #8
CDAA20D2FF 130 JSR CHROUT
132 ;
134 ;
CDAD20C2CD 136 CH JSR CHARIO
CDB0C65C 138 DEC CCHAR
CDB2D0F9 140 BNE CH
CDB4 A901 142 LDA #1
CDB6855D 144 STA DRAPD
CDB820C2CD 146 JSR CHARIO
CDBB20C3FF 148 JSR CLOSE
CDBE20C0FF 149 JSR CLRCHN
CDC1 60 150 RTS
151 ;
152 ;
153 ;U*****S.P. CHARIO*****
CDC2 A900 154 CHARIO LDA #0
CDC4 8558 156 STA PDOT
CDC6 A900 158 LDA #0
CDC8 8559 160 STA PDOTH
CDCA855A 162 STA CTROU
164 ;
CDCC20FD0D 166 SE JSR SEPTET
CDCF E658 168 INC PDOT
CDD1D002 170 BNE TEST
CDD3 E659 172 INC PDOTH
CDD5 A940 174 TEST LDA #4
CDD7 C558 176 CMP PDOT
CDD9D0F1 178 BNE SE
CDDB A901 180 LDA #1
CDDD C559 182 CMP PDOTH
CDDFD0EB 184 BNE SE
186 ;
CDE1 A90D 188 LDA #13 ;RETOUR CHARIOT
CDE320D2FF 190 JSR CHROUT
192 ;
CDE6 A907 194 LDA #00000111
CDE8 24FB 196 BIT PSET
CDEA F003 198 BEQ MUL
CDEC C6FB 200 DEC PSET
CDEE 60 201 RTS
CDEF 38 202 MUL SEC
CDF0 85FB 204 LDA PSET
CDF2 E939 206 SBC #57
CDF4 35FB 208 STA PSET
CDF6 85FC 208 LDA PSETH
CDF8 E901 210 SBC #1
CDFA 85FC 211 STA PSETH
CDFC 60 212 RTS
214 ;
216 ;*****S.P. SEPTET***
CDFD203FCE 218 SEPTET JSR SEPSUI
CE00 C980 220 CMP #10000000
CE02 D009 222 BNE NN
CE04E65A 224 INC CTROU ;CTROU REMIS A 128
CE06 D004 226 BNE RET ;SI DEPASSEMENT
CE08 A940 228 LDA #40
CE0A855A 230 STA CTROU ;
CE0C 60 232 RET RTS
234 ;
CE0D 48 236 NN PHA ;SAUVE. LE SEPTET
CE0E855A 238 LDA CTROU
CE10 C900 240 CMP #0
CE12F018 241 BEQ ENVOIE
    
```



```

CE14 A91B 254 LDA #27 ;MESSAG DOT POS.
CE1620D2FF 256 JSR CHROUT ;
CE19 A910 258 LDA #16 ;
CE1B20D2FF 260 JSR CHROUT ;
CE1E A559 262 LDA P00TH ;
CE20C002FF 264 JSR-CHROUT ;
CE23 A558 266 LDA P00T ;
CE2520D2FF 268 JSR CHROUT ;
CE28 A900 270 LDA #0 ;CTROU=0
CE2A855A 272 STA CTROU ;
CE2C 68 274 ENVOIE PLA ;ON ENVOIE LE SEPTET
CE2D20D2FF 276 JSR CHROUT ;
CE30 60 278 RTS ;
280 ;
282 ;
CE31 A980 284 PEU LDA #128 ;ON ENVOIE LES TROUS
CE3320D2FF 286 JSR CHROUT ;
CE36C65A 288 DEC CTROU ;
CE38 D0F7 290 BNE PEU ;
CE3A 68 292 PLA ;ON ENVOIE LE SEPTET
CE3B20D2FF 294 JSR CHROUT ;
CE3E 60 296 RTS ;
298 ;
300 ;
302 ;*****S.P. SEPSUI*****
CE3F A5FB 304 SEPSUI LDA PSET ;
CE41 85FD 306 STA PPSET ;
CE43 A5FC 308 LDA PSETH ;
CE45 85FE 310 STA PPSETH ;
CE47 A200 312 LDX #0 ;INITIALISE SEPTET A 0
CE49 A000 314 LDY #0 ;POUR AD. INDEXE
CE4B A907 316 LDA #7 ;COMPTEUR DE BOUCLE
CE4D855E 318 STA CPIX ;
320 ;
322 ;
CE4F EA 324 BOUCLE NOP ;
CE50 18 325 CLC ;
CE51 78 326 SET ; ;64 K RAM
CE52 A501 327 LDA #01 ;
CE54 48 328 PHA ;
CE55 A934 329 LDA #52 ;
CE57 8501 339 STA #01 ;
CE59A5FF 340 LDA BISET ;
CE5B 31FD 341 AND (PPSET),Y ;TEST UN PIXEL
CE5DF001 343 BEQ RIEN ;SI PIXEL ETEINT
CE5F 38 345 SEC ;
CE60 68 347 RIEN PLA ;
CE61 8501 349 STA #01 ;
CE63 58 350 CLI ;
CE64 8A 351 TXA ;
CE65 6A 353 ROR ;
CE66 AA 355 TAX ;
357 ;

```

```

CE67C65E 359 DEC CPIX ;ENCORE DES PIXELS A TESTER
CE69F006 361 BEQ FINPIX ;
CE6B 207ECE 363 JSR POS ;POINTAGE OCTET SUIVANT
CE6E 38 365 SEC ;BRANCHE. FORCE
CE6FB0DE 367 BCS BOUCLE ;
369 ;
CE71 209ACE 371 FINPIX JSR PSS ;POINTAGE SEPTET SUIVANT
CE74 38 373 SEC ;AJOUTE 128 AU SEPTET
CE75 8A 375 TXA ;
CE76 6A 377 ROR ;
379 ;
CE77A65D 381 LDX DRAP0 ;TEST DERNIER CHARIOT
CE79 F002 383 BEQ FINSS ;
CE7B 290F 385 AND #210001111 ;
CE7D 60 387 FINSS RTS ;
389 ;
391 ;
393 ;*****S.P. POINTAGE OCTET SUIVANT*****
CE7E 18 395 POS CLC ;
CE7F E6FD 397 INC PPSET ;
CE81 D002 399 BNE RETNU ;
CE83 E6FE 401 INC PPSETH ;
CE85 A907 403 RETNU LDA #200000111 ;ADRESSE MULTIPLE DE HUIT
CE87 24FD 405 BIT PPSET ;
CE89 F001 407 BEQ MULHUI ;
CE8B 60 409 RTS ;
411 ;
CE8C 18 413 MULHUI CLC ;
CE8D A938 415 LDA #56 ;
CE8F 65FD 417 ADC PPSET ;
CE91 85FD 419 STA PPSET ;
CE93 A901 421 LDA #1 ;
CE95 65FE 423 ADC PPSETH ;
CE97 85FE 425 STA PPSETH ;
CE99 60 427 RTS ;
429 ;
431 ;
433 ;*****S.P. POINTAGE SEPTET SUIVANT*****
CE9AA5FF 435 PSS LDA BISET ;
CE9C 4A 437 LSR ;
CE9D85FF 439 STA BISET ;
CE9FB001 441 BCS DEPASS ;
CEA1 60 443 RTS ;
445 ;
CEA2 6A 447 DEPASS ROR ;
CEA385FF 449 STA BISET ;
CEA5 18 451 CLC ;
CEA6 A908 453 LDA #8 ;
CEA8 65FB 455 ADC PSET ;
CEAA 85FB 457 STA PSET ;
CEAC 9002 459 BCC FINPS ;
CEAE E6FC 461 INC PSETH ;
CEB0 60 463 FINPS RTS ;

```

## COPIE RAPIDE

```

1 REM SAVE"@DCOP52608".8
90 :
100 REM COPIE RAPIDE DE L'ECRAN HAUTE RESOLUTION
110 REM PAR R. TISSERAND (1985)
112 :
113 :
130 FORF=52608 TO 52913 STEP 48
140 CS=0
150 FORN=FTOF+47:READA:POKE N,A:CS=CS+A:NEXT N
160 PRINT CS,:NEXT F

```

```

52608 DATA169,000,133,251,169,224,133,252
52616 DATA169,128,133,255,169,028,133,092
52624 DATA169,000,133,093,169,000,133,183
52632 DATA133,185,169,004,133,184,133,186
52640 DATA032,192,255,162,004,032,201,255
52648 DATA169,008,032,210,255,032,194,205
52649 REMCHECKSUM= 6683 DE 52608 A 52655
52656 DATA198,092,208,249,169,001,133,093
52664 DATA032,194,205,032,195,255,032,204
52672 DATA255,096,169,000,133,088,169,000
52680 DATA133,089,133,090,032,253,205,230
52688 DATA088,208,002,230,089,169,064,197
52696 DATA088,208,241,169,001,197,089,208
52697 REMCHECKSUM= 6615 DE 52656 A 52703
52704 DATA235,169,013,032,210,255,169,007
52712 DATA036,251,240,003,198,251,096,056
52720 DATA165,251,233,057,133,251,165,252
52728 DATA233,001,133,252,096,032,063,206
52736 DATA201,128,208,009,230,090,208,004
52744 DATA169,064,133,090,096,072,165,090
52745 REMCHECKSUM= 6701 DE 52704 A 52751
52752 DATA201,000,240,024,169,027,032,210
52760 DATA255,169,016,032,210,255,165,089
52768 DATA032,210,255,165,088,032,210,255
52776 DATA169,000,133,090,104,032,210,255
52784 DATA096,169,128,032,210,255,198,090
52792 DATA208,247,104,032,210,255,096,165
52793 REMCHECKSUM= 6829 DE 52752 A 52799

```

READY.

```

52800 DATA251,133,253,165,252,133,254,162
52808 DATA000,160,000,169,007,133,094,234
52816 DATA024,120,165,001,072,169,052,133
52824 DATA001,165,255,049,253,240,001,056
52832 DATA104,133,001,088,138,106,170,198
52840 DATA094,240,006,032,126,206,056,176
52841 REMCHECKSUM= 6030 DE 52800 A 52847
52848 DATA222,032,154,206,056,138,106,166
52856 DATA093,240,002,041,143,096,024,230
52864 DATA253,208,002,230,254,169,007,036
52872 DATA253,240,001,096,024,169,056,101
52880 DATA253,133,253,169,001,101,254,133
52888 DATA254,096,165,255,074,133,255,176

```

READY.

# MODEMS

## RS 232 ET LA FIN DE L'ÎLE DESERTE

**Pour ne pas laisser votre Commodore 64 tout seul, pour lui permettre d'échanger de menus propos avec d'autres Commodore 64 sans disquettes ou cassettes interposées, de faire la casette avec d'autres ordinateurs de marques différentes, et même d'échanger des fichiers (en ASCII standard) ou des programmes (en Basic standard)...**

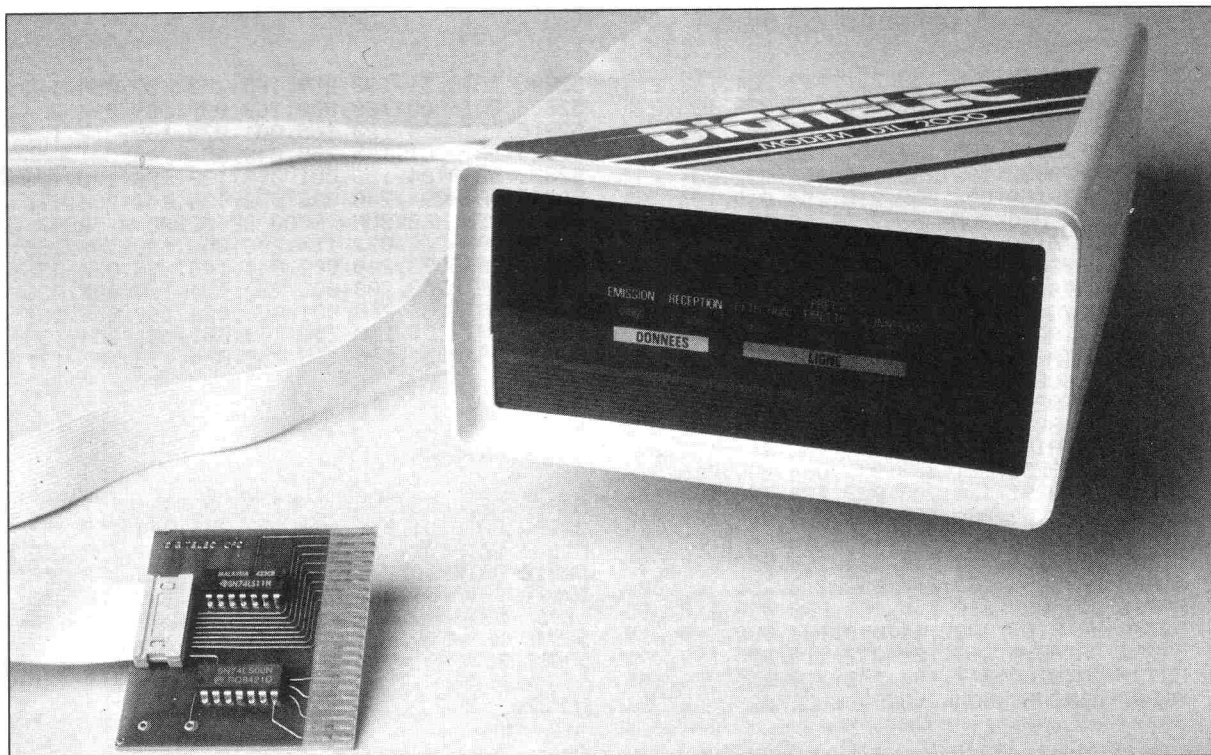
**D**ans les mois qui suivirent la création du VIC-20 apparaissait dans les catalogues un petit accessoire du nom de VIC 1011A, qui a laissé, disons-le tout net, le « bon peuple » passablement indifférent. On se l'arrache aujourd'hui, au rythme de parcimonieuses importations. Il ressemble à une cartouche, il a la couleur d'une cartouche, mais n'est pas une cartouche.

Il se branche sur le port utilisateur, il est muni d'une prise Cannon 25 broches, il est livré dans un élégant sachet de plastique transparent, avec un manuel d'utilisation de dix pages, et c'est l'interface R232C du VIC-20 et du Commodore 64. Avec la découverte, par la France profonde, de la télématique, et pour ceux qui savent que derrière « télé-informatique » se cachent des modems, cet astucieux petit accessoire connaît une nouvelle vogue. Comment s'en sert-on ?

Pour le C. 64, le bus RS232 est un périphérique

comme un autre : il s'ouvre avec OPEN, se ferme avec CLOSE et se gère avec soin au moyen des classiques CMD, PRINT#, GET# et INPUT#. Ah, pendant que j'y pense, pour se servir d'un périphérique RS232, il faut ouvrir un canal RS232. Quand on ouvre un canal RS232, c'est comme si on faisait un CLR, on perd toutes les variables en mémoire. Vous voilà prévenu.

Bon, alors, procédons par ordre. Mettons qu'on veuille se servir d'une imprimante RS232 ou d'un modem. On a intérêt à ouvrir le canal en début de programme. Le port RS232 s'est vu attribuer par Commodore le numéro de périphérique 2, et s'ouvre avec une adresse secondaire nulle et un nom de fichier de deux caractères (lourds de conséquences), puisque ce sont eux qui, par leur valeur ASCII, fixent les différents paramètres de communication. Cette ouverture se fait sous la forme :



C. 64 Spécial - H.S. 70 - Mai 85





OPEN#2,2,0,CHR\$(A)+CHR\$(B)

Les différents paramètres sont d'abord pour A, la vitesse de transmission exprimée en caractères par seconde, ou CPS (ne dites surtout pas bauds, vous vous feriez écharper par les puristes !), puis la longueur des mots transmis, exprimée en bits, puis le nombre de bits d'arrêt. Il y a d'autre part pour B, le nombre de lignes intervenant dans la communication, puis le mode de communication (full duplex ou half duplex), l'utilisation (ou non) d'une parité, et les modalités de cette parité.

Dans la pratique, à quoi cela sert-il ? Cela ne fait pas le café, mais cela permet d'ouvrir son petit univers à d'autres périphériques que ceux qu'offre Commodore. On peut y brancher un modem ordinaire, un Buzzbox, par exemple, et communiquer avec d'autres ordinateurs par téléphone (entre gens d'une même circonscription téléphonique si on hésite à risquer l'infarctus à l'arrivée de la note), on peut se brancher sur un serveur, on peut accéder à TRANSPAC, et, sans le moindre téléphone, mais avec ce que l'on appelle pompeusement et communément un *modem nul*, entendez deux prises et un bout de câble ; on peut aussi communiquer avec un autre ordinateur de marque différente, posé sur la même table, pourvu qu'il ait, lui aussi, le standard RS232.

### Même avec un modem « nul », attention à la note

Permettez-moi de vous dire que, passés les premiers émerveillements du modem et les premières notes de téléphone, c'est peut-être ce qui m'a procuré le plus de plaisir, voire le plus de soulagement. Un programme écrit en BASIC standard, sans PEEKs ni POKEs, sans acrobaties graphiques ni sonores, le bon programme que vous vous êtes mitonné et que vous avez peaufiné avec les années n'est plus cantonné à une machine unique. Vous pouvez le sortir de votre ancien ordinateur, ou de celui de « papa », vous pouvez en faire profiter les copains. Les quelques particularismes à corriger sont plus facile à

modifier sur place qu'une dizaine de lignes. Autant éviter de tout refrapper au clavier ! Grosso et modo, avec modem ou sans, la liaison RS232, c'est la fin de l'île déserte.

### Choisissez le bon modem et branchez-vous au port

Certains modems se branchent directement sur le port utilisateur du C. 64. Ils présentent souvent l'avantage d'être livrés avec un logiciel de communication. Inconvénient : on ne dispose pas du port RS232. Les autres fonctionnent avec n'importe quel ordinateur, mais nécessitent l'interface VIC 1011A. Ils sont en général livrés sans logiciel de communication, mais l'ensemble forme un outil bien plus souple.

Attention également au choix du modem. Deux standards de communication sont actuellement utilisés en France : le V21 (ou V24), qui fonctionne à 300 bauds, et dont la lenteur permet d'utiliser sans problème des lignes téléphoniques de qualité moyenne, et le V23, en 1200 bauds à la réception et 75 seulement en émission, qui donne accès aux services Teletel, pourvu, là encore, que l'on dispose d'un programme de communication, capable de transformer les codes Vidéotext en codes Commodore, ce qui n'est pas une mince affaire. Nous l'avons résolue dans le numéro 70 de *l'Ordinateur Individuel* ■

François Chevalier  
avec Alain Mariatte

NOTE : A lire absolument : le dossier communication de *l'Ordinateur Individuel* n° 64 (p. 180 et suiv.), le dossier Minitel paru dans le n° 65 (p. 155 et suiv.), le programme de communication du numéro 70, p. 147 à 149.

### BRANCHEMENTS RAPIDES

Les serveurs vont et viennent. On se branche. Ça marche, ça ne marche pas. En voici quelques-uns qui fonctionnaient en ASCII au moment où les rotatives tournaient, les compositeurs composaient et les lecteurs communiquaient.

**CALVADOS** : Le seul vrai serveur multi-entrées. Tout le monde peut y accéder par TRANSPAC, moyennant un abonnement et, comme pour les autres, les taxes téléphoniques. Tél. : 16 (1) 601.91.00 ou 16 (1) 613.91.55, puis 175.040.781.

**EVEREST** : Un très bon CBBS à la française. Une foule de possibilités en plus de la messagerie (P.A., bulletin, troc, test de matériel, jeux, presse informatique...). Gratuit. Tél. : 16 (1) 367.19.80.

**FUTURA** : Attention à l'aurografe ! Plutôt intéressant, se visite gratuitement. Quelques services payants. Tél. : 16 (1) 500.30.15.

**GUFINET** : Téléchargement ! Gratuit en messagerie, payant pour tout le reste qui vaut le coup ! Non réservé aux CP/Mistes. Tél. : (1) 565.10.09.

**SCHTHROUMF** : Il existe, mais on a du mal à se brancher. Tél. : 16 (1) 876.45.67.

**SPACECOM** : C'est le serveur dit « Commodore ». Paraît cher pour les services offerts. Tél. : 16 (37) 82.40.18.

**TELEPOM** : Rigolo-Moreno. Tél. : 16 (1) 531.57.25.

# GAMMES

## SID CHIP : LA PUCE MÉLOMANE

**Les commentateurs n'ont cessé d'affirmer que le C. 64 possédait des possibilités sonores inouïes. « Inouïes », c'est-à-dire « jamais entendues ». Et c'est un peu vrai : bien des possesseurs de C. 64 n'ont jamais entendu le son de leur machine, sauf dans certains programmes de démonstration ou de jeux. Il est vrai que la programmation des sons du C. 64 est un peu compliquée mais pas impossible. La preuve !**



**N**ous nous limiterons pour commencer à des sons simples produits par le circuit intégré propre à Commodore, le SID (Sound Interface Device) 6581. Véritable synthétiseur dont la sortie commande le son du téléviseur, il génère trois voix indépendantes et admet une entrée extérieure sur laquelle il effectue des filtrages.

Le processeur voit le SID comme un ensemble de cases mémoire qui forment ses registres. Vingt-neuf registres d'adresses au total allant de 54272 à 54300. On obtient des sons en écrivant des données dans les différents registres du SID. Rappelons que pour écrire dans une case mémoire en Basic, on

utilise l'instruction POKE. Ainsi, POKE 54296, 15 va influencer sur les sons produits (cette instruction met le volume sonore au maximum).

Le SID comporte quatre séries de registres : trois séries de sept registres jouent des rôles

parallèles, respectivement pour les voix (et voix !) 1, 2 et 3, et une série de registres globaux. Nous n'en utiliserons que certains, en nous limitant d'abord à la voix 1. Ensuite, le seul registre général que nous utiliserons sera celui du volume (54296). On y mettra une valeur de 0 (silence) à 15 (volume maximal).

Il nous faut maintenant une méthode pour désigner les adresses de ces registres. Il sera plus parlant d'utiliser des noms de variables plutôt que les nombres eux-mêmes. D'autre part, on construira les adresses à partir de l'adresse de base SID :  $S = 54272$ .

Ainsi, pour le volume, nous aurons :  $V = S + 24$ . Passons en revue les autres registres :

**FB = S :** c'est la partie basse du nombre qui détermine la fréquence du son, donc sa hauteur.

**FH = S + 1 :** c'est la partie haute du nombre qui détermine la fréquence. On a donc une précision de 16 bits pour les fréquences.

**F = S + 4 :** détermine la forme d'onde, c'est-à-dire le timbre, et lance le son. Les valeurs à mettre sont :

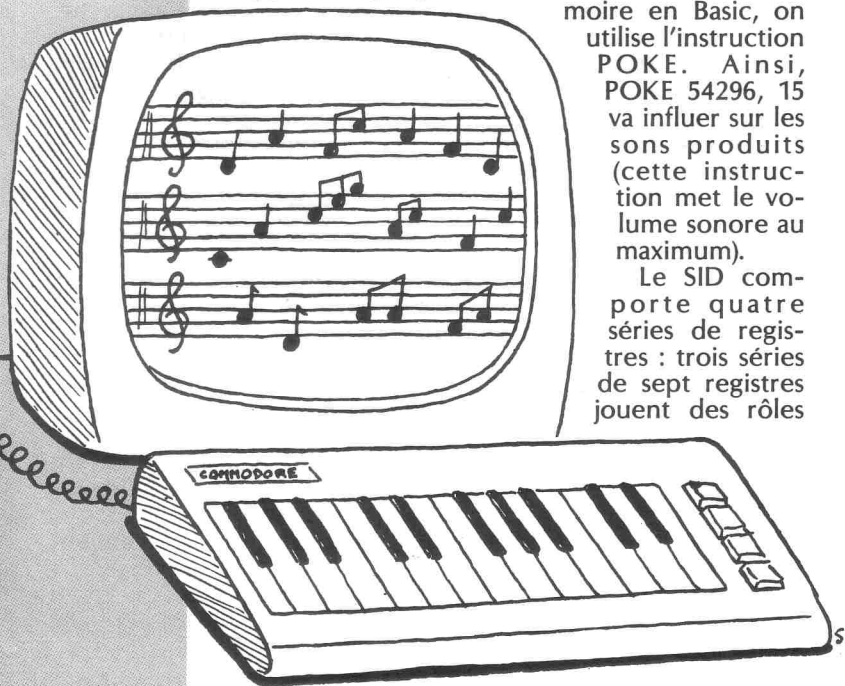
- FF = 16 onde triangulaire
- = 32 dent de scie
- = 64 rectangulaire
- = 128 bruit

On fait POKE F, FF + 1 pour lancer le son et POKE F, FF pour commencer et arrêter le son.

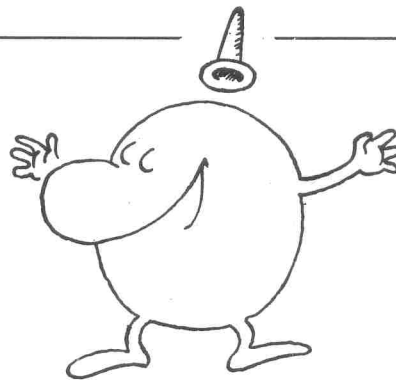
**AD = S + 5 :** détermine la vitesse d'attaque et celle de descente du son (voir ci-dessous la question de l'enveloppe).

**SR = S + 6 :** détermine le taux de soutien et la vitesse de relaxation (enveloppe).

L'enveloppe, définie par ces deux derniers registres, est, avec le timbre, à la source des capacités d'imitation des instruments de musique. L'enveloppe est la courbe, en quatre épisodes, qui suit l'intensité du son.





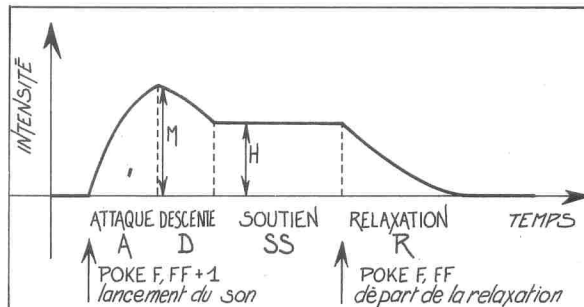


## PROGRAMME 1

```

10 REM DEFINIT LES ADRESSES
20 S=54272:V=S+24:FB=S:FH=S+1:F=S+4:AD=S+5:SR=
S+6
30 REM VIDE LE SID
40 FOR I=0 TO 28:POKES+I,0:NEXT
50 REM FIXE VOLUME ET ENVELOPPE
60 POKEV,15:POKEAD,52:POKESR,168
70 REM TIMBRE=DENT DE SCIE, NOTE=LA-3
80 FF=32:POKEFB,195:POKEFH,28
90 REM LANCE LE SON
100 POKEF,FF+1
110 REM DELAI 1 S
120 FOR T=1 TO 1000:NEXT
130 REM ARRETE LE SON
140 POKEF,FF
READY.

```



La valeur mise en AD est  $16 \times A + D$  avec A de 0 (durée de l'attaque 2 ms) à 15 (8 s) et D de 0 (durée de la descente 6 ms) à 15 (24 s).

La valeur mise en SR est  $16 \times SS + R$  avec SS de 0 à 15 qui définit la fraction H/M et R de 0 (durée de la relaxation 6 ms) à 15 (24 s).

Pour le programme que nous allons écrire, nous prendrons des valeurs moyennes : A = 3, D = 4, SS = 10 et R = 8, soit POKE AD, 52 et POKE SR, 168. Vous pouvez essayer d'autres valeurs : dans ce domaine, il faut expérimenter.

Voici le scénario des opérations à effectuer pour obtenir un son.

- 1 - Vider tous les registres du SID pour qu'il n'y reste pas de valeurs intempestives :  
POKE = 0 TO 18 : POKES + I, 0 : NEXT
- 2 - Fixer le volume :  
POKE V, 15
- 3 - Déterminer l'enveloppe :  
POKE AD, 52 = POKE SR, 168
- 4 - Déterminer la fréquence de la note (par exemple : la - 3) : POKE FB, 195 : POKE FH, 28
- 5 - Lancer la note, en choisissant en même temps son timbre : FF = 32 : POKE F, FF + 1
- 6 - Délai de la durée de la note (par exemple : environ 1 seconde)  
FOR T = 1 TO 1000 : NEXT
- 7 - Arrêter la note : POKE F, FF

## PROGRAMME 2

```

10 REM DEFINIT LES ADRESSES
20 S=54272:V=S+24:FB=S:FH=S+1:F=S+4:AD=S+5:SR=
S+6
30 REM VIDE LE SID
40 FOR I=0 TO 28:POKES+I,0:NEXT
50 REM FIXE VOLUME ET ENVELOPPE
60 POKEV,15:POKEAD,52:POKESR,168
70 REM TIMBRE=DENT DE SCIE
80 FF=32
82 READ XB,XH
84 IF (XB=0) AND (XH=0) THEN END
86 POKEFB,XB:POKEFH,XH
90 REM LANCE LE SON
100 POKEF,FF+1
110 REM DELAI 0.1 S
120 FOR T=1 TO 100:NEXT
130 REM ARRETE LE SON
140 POKEF,FF
150 GOT082
1000 DATA 24,17,50,19,140,21,211,22,160,25,195
,28,72,32,52,34
1010 DATA 52,34,72,32,195,28,160,25,211,22,140
,21,50,19,24,17,0,0
1020 REM GAMME MONTANTE PUIS DESCENDANTE
READY.

```

Il ne manque que de définir les adresses au début et le tour est joué (programme 1).

Que faut-il pour jouer la gamme ? Eh bien repasser pour chaque note sur les instructions 70 à 140 du programme 1. Le mieux est de grouper les valeurs des notes en DATA. Bien sûr, vous pourrez jouer une autre mélodie que la gamme (programme 2).

Et voilà ! Pas plus difficile que cela. Vous noterez l'astuce de terminer les DATA par 0,0 en ligne 1010.

Il est évident que le C. 64 offre des possibilités plus riches et plus complexes, notamment dans le domaine des accords (plusieurs voix simultanées). A vous de continuer...

Daniel-Jean David



### LES REGISTRES DU SYNTHETISEUR SONORE DU C. 64

	H EXA		DEC	BASE +
VOIX 1	D400	FREQUENCE BASSE (F0-F7) VOIX 1		54272 +0
	D401	FREQUENCE HAUTE (F8-F15)		54273 +1
	D402	RAPPORT CYCLIQUE BAS (PW0-PW7)		54274 +2
	D403	RPT CYC HAUT (PW8-PW11)		54275 +3
	D404	bruit rect scie tri FORME D'ONDE	test ring sync porte MOD	54276 +4
	D405	ATTAQUE	DESCENTE	54277 +5
D406	SOUTIEN	RELAXATION	54278 +6	
VOIX 2	D407	FREQUENCE BASSE (F0-F7) VOIX 2		54279 +7
	D408	FREQUENCE HAUTE (F8-F15)		54280 +8
	D409	RAPPORT CYCLIQUE BAS (PW0-PW7)		54281 +9
	D40A	RPT CYC HAUT (PW8-PW11)		54282 +10
	D40B	bruit rect scie tri FORME D'ONDE	test ring sync porte MOD	54283 +11
	D40C	ATTAQUE	DESCENTE	54284 +12
D40D	SOUTIEN	RELAXATION	54285 +13	
VOIX 3	D40E	FREQUENCE BASSE (F0-F7) VOIX 3		54286 +14
	D40F	FREQUENCE HAUTE (F8-F15)		54287 +15
	D410	RAPPORT CYCLIQUE BAS (PW0-PW7)		54288 +16
	D411	RPT CYC HAUT (PW8-PW11)		54289 +17
	D412	bruit rect scie tri FORME D'ONDE	test ring sync porte MOD	54290 +18
	D413	ATTAQUE	DESCENTE	54291 +19
D414	SOUTIEN	RELAXATION	54292 +20	
GENERAUX	D415	FREQ FILTRE BASSE FC0-FC2		54293 +21
	D416	FREQ FILTRE HAUTE FC3-FL10		54294 +22
	D417	RESONANCE 0-3	ext 3 2 1 ACTIVE FILTRE VOIX	54295 +23
	D418	silence haut bande bas VOIX 3 FILTRE PASSE	VOLUME VO-V3	54296 +24
	D419	POTENTIOMETRE X PX0-PX7		54297 +25
	D41A	POTENTIOMETRE Y PY0-PY7		54298 +26
	D41B	OSC 3 / HASARD 00-07		54299 +27
	D41C	ENVELOPPE 3 E0-E7		54300 +28

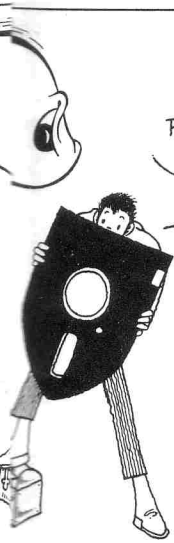
### MONSTRES ET TRÉSORS

**Ne vous faites pas dévorer par les monstres, ni effrayer par les DATA des lignes 10 à 330 : elles ne servent qu'à définir les lutins. Vous pouvez vous déplacer à l'aide du joystick 2 ou avec les touches A, Z, N, M et espace.**

```

5 REM MONSTRES ET TRESORS POUR CBM 64
6 REM PAR CYRIL CAMBIEN - 04/1984 -
7 REM -----
8 REM CODAGE DES GRAPHISMES
10 DATA 0,255,0,3,129,192,7,0,224,12,0,4
8,25,195,152,27
20 DATA 102,216,49,195,140,48,0,12,48,0,
12,51,0,204,49,129
30 DATA 140,24,195,24,24,60,24,12,0,48,7
,0,224,195,129,195
40 DATA 96,255,6,48,0,12,31,255,248,0,60
,0,0,24,0,0
50 DATA 0,24,0,0,0,60,0,31,255,248,48,0,12
,96,255,6,195
60 DATA 129,195,7,0,224,12,0,48,24,60,24
,24,195,24,49,129
70 DATA 140,51,0,204,48,0,12,48,0,12,49,
195,140,27,102,216
80 DATA 25,195,152,12,0,48,7,0,224,3,129
,192,0,255,0,0
90 DATA 0,0,32,0,0,48,3,252,24,14,7,12,2
8,3,132,49
100 DATA 8,196,99,140,100,98,134,100,195
,131,52,193,1,54,192,1
110 DATA 55,193,1,54,195,131,52,98,134,1
00,99,140,100,49,8,196
120 DATA 28,3,132,14,7,12,3,252,24,0,0,4
8,0,0,32,159
130 DATA 4,0,0,12,0,0,24,63,192,48,224,1
12,33,192,56,35
140 DATA 16,140,38,49,198,38,97,70,44,19
3,195,108,128,131,236,128
150 DATA 3,108,128,131,44,193,195,38,97,
70,38,49,198,35,16,140
160 DATA 33,192,56,48,224,112,24,63,192,
12,0,0,4,0,0,157
170 DATA 0,127,0,0,255,128,0,201,128,0,2
01,128,0,255,128,0
180 DATA 227,128,0,221,128,0,127,0,0,28,
0,31,255,252,63,255
190 DATA 254,60,255,158,56,255,142,56,25
5,142,56,247,142,124,255,159
200 DATA 109,227,219,1,227,192,3,193,224
,3,193,224,7,193,240,157
    
```





# REFLEXES

## DISTRAIRE SON COMMODORE

**Partir à la découverte d'un fabuleux trésor, atterrir en catastrophe dans une contrée inconnue, faire clignoter son écran, le coup paraît usé. Mais il y a mille et une manières d'exploiter les ressources du C. 64. Vive la ré-creative !**

```

210 DATA 0,127,0,0,221,128,0,255,128,0,2
47,128,0,99,0,0
220 DATA 54,0,0,28,0,0,28,0,0,28,0,0,56,
0,0,112
230 DATA 0,0,224,0,1,199,192,3,136,96,7,
16,48,7,16,48
240 DATA 7,16,48,7,8,112,3,128,224,1,255
,192,0,255,128,157
250 DATA 0,255,128,1,255,192,3,28,96,3,2
8,96,3,182,224,3
260 DATA 247,224,1,227,192,112,255,135,1
6,99,4,24,93,12,124,62
270 DATA 31,14,28,56,7,255,240,3,190,224
,0,221,128,0,190,128
280 DATA 0,221,128,0,247,128,1,227,192,1
,193,192,3,193,224,206
290 DATA 1,255,255,3,0,7,3,0,7,6,0,13,6,
0,13,12
300 DATA 0,27,15,255,254,24,0,24,48,0,56
,96,0,120,255,255
310 DATA 216,224,3,152,193,193,152,193,6
5,152,193,193,152,192,1,152
320 DATA 192,1,152,192,1,176,192,1,224,2
24,3,192,255,255,128,206
330 DATA 132
335 REM COULEURS ET NOMS DES MONSTRES
340 DATA 6,ZOMBIES,5,SERPENTS,1,SQUELETTE
S
400 POKE56,62:CLR
499 REM INITIALISATION
500 FORI=15872TO16384:READA:POKEI,A:NEXT
510 FORI=0TO2:READC(I):READN(I):NEXT
515 GOSUB6000:REM INSTRUCTIONS
520 V=53248
530 POKEV+23,255:POKEV+29,255
550 TA=10:X=110:T=0:M=0
590 POKE198,0:INPUT"QDIFFICULTE (1-3)
DI
591 IFDI<1ORDI>3GOTO590
595 POKEV+21,255
600 GOSUB1000,GOSUB1200,GOSUB1300
605 A=PEEK(V+30)
610 V|=5
699 REM PROGRAMME CENTRAL
700 J=1:IFPEEK(563207K)>127THENGOSUB2100:
GOTO710
705 IFPEEK(203)>64THENGOSUB2000
710 GOSUB4000
750 IFY<>175GOTO800
755 A=INT(X/8)+40*INT(Y/8)+1024
760 IFPEEK(A)=32THENIFPEEK(2040)=248GOTO
600
800 A=PEEK(V+30)
810 IF(ORANDI)=0GOTO700

```

```

820 IF(ORANDI6)GOTO5000
825 REM MORT DU JOUEUR
830 FORI=1TO4:FORJ=248TO251:POKE2040,J:IF
ORR=1TO50:NEXTK,J,I
835 POKEV+21,0
840 PRINT"ENVOUS ETES MORT !!!"
845 S$="":IFT>1THENS$="S"
850 PRINT"PRINT VOUS AVEZ VOULET TRESOR"
S$
855 S$="":IFM>1THENS$="S"
860 PRINT"PRINT ET VOUS AVEZ TUE M MONST
RE S$
870 PRINT"PRINT APPUYEZ SUR SHIFT POUR F
AIRE UNE PARTIE
880 WAIT653,1:GOTO530
1000 REM DESSIN DE LA PIECE
1001 PRINT"EN CHR$(142),
1005 POKE53281,15
1020 PRINT"R":FORI=1TO30:PRINT"O":NEXT
I:PRINT"Y"
1021 POKE211,TA:PRINT"R"
1023 FORI=1TO20
1030 PRINT"R":TAB(29)"O
1035 NEXT
1100 PRINT"R":
1110 FORI=1TO30:PRINT"O":NEXTPRINT"R"
1120 TA=RND(1)*20+3:POKE211,TA:PRINT"R"
ENVOUS AVEZ TUE M MONSTRE
1130 RETURN
1200 REM CHOIX DE L'ADVERSAIRE
1201 REM POSITIONNEMENT DES ADVERSAIRES
ET DU TRESOR
1210 AD=INT(RND(1)*3)
1220 FORI=2041TO2043:POKEI,252+AD:NEXT
1225 POKE2044,255:POKEV+43,7
1230 FORI=0TO2TOV+95STEP2
1240 POKEI,RND(1)*180+30
1245 POKEI+1,RND(1)*15+110
1250 NEXT
1255 FORI=0TOV+42:POKEI,C(AD):NEXT
1260 POKE214,23:PRINT"Q LES "N$(AD)" A
TTAQUENT !!!"
1290 RETURN
1300 REM POSITIONNEMENT DU JOUEUR
1310 Y=60:IFTHENY=175
1320 POKE2040,248:P=248:UX=0:UY=1:A$="2
1330 POKEV,X-POKEV+1,Y
1340 POKEV+39,10
1390 RETURN
2000 REM DEPLACEMENT DU JOUEUR
2010 A=PEEK(203)
2020 IFA=10THENPOKE2040,249:UX=0:UY=-1:A
$="1":IFY>60THENY=Y-VI
2030 IFA=12THENPOKE2040,248:POKE2040,248

```

```

1 UX=0:UY=1:A$="2" IFY<175THENY=Y+U[
2040 [FA=36THENPOKE2040,250 UX=1:UY=0:A$
="1": [FX<210THENX=X+U[
2050 [FA=39THENPOKE2040,251 UX=-1:UY=0:A$
="3": [FX>30THENX=X-U[
2060 POKEU,X:POKEU+1,Y
2070 [FA=60THEN3000
2080 RETURN
2100 REM DEPLACEMENT DU JOUEUR (JOY)
2110 A=PEEK(56320)
2120 [F(AAND1)=0THENP=249:UX=0:UY=-1:A$="
1": [FY>60THENY=Y-U[
2130 [F(AAND2)=0THENP=248:UX=0:UY=1:A$="
2": [FY<175THENY=Y+U[
2140 [F(AAND8)=0THENP=250:UX=1:UY=0:A$="
+": [FX<210THENX=X+U[
2150 [F(AAND4)=0THENP=251:UX=-1:UY=0:A$="
3": [FX>30THENX=X-U[
2160 POKEU,X:POKEU+1,Y:POKE2040,P
2170 [F(AAND16)=0THENGOSUB3000
2180 RETURN
3000 REM TIR DU JOUEUR
3005 PRINT "Sa

```

```

3010 TX=INT(X/8)+3*UX
3020 TY=INT(Y/8)-5+3*UY
3022 IFTY<0ORTY>18ORTX<1ORTX>28THENRETURN
3030 A=i024+TX+TY*40+40
3035 IFPEEK(A)<>32THENRETURN
3040 POKE211,TX:POKE214,TY:PRINT"Q A$
3045 A=PEEK(U+31):FORI=1TO10:NEXT
3050 A=PEEK(U+31):IFA<>0THEN3100
3080 PRINT "A";
3090 TX=TX+UX:TY=TY+UY:GOTO3030
3100 A=INT(LOG(A)/LOG(2)): [FA=0THEN3080
3110 POKEU+2*A,0:POKEU+1+2*A,0
3180 PRINT "A":M=M+1:A=PEEK(U+30)
3190 RETURN
4000 REM DEPLACEMENT DES MONSTRES
4010 FORI=U+2TOU+7STEP2
4015 IFPEEK(I)=0GOTO4080
4020 POKEI,PEEK(I)+3-INT(RND(1)*6)+(PEEK
K(I)>X)-(PEEK(I)<X)*DI
4030 J=I+1:POKEJ,PEEK(J)+3-INT(RND(1)*6)
+(PEEK(J)>Y)-(PEEK(J)<Y)*DI
4040 IFPEEK(I)>200THENPOKEI,200

```



**SOS ATERRISSAGE**

**Là on dirige l'engin avec les touches A, Z, N et M. Rejoignez votre base entier !**

```

0 GOTO501 >>> SOS ATERRISSAGE PAR BENOIT
ROLAND <<<
1 REM TOUCHES A,Z,M & N POUR DIRIGER
L'ENGIN
2 G=INT(RND(1)*35):PRINT"s_
3 FORI=1984TO2024
4 POKEI,102:POKEI+54272,1
5 NEXT
9 J=200:K=0
50 N=1441
110 GOTO300
111 B=INT(RND(1)*A)
112 POKEN+41,32
120 POKEN,152:POKEN+54272,1
121 IFPEEK(N+40)=102THEN700
122 IFPEEK(N+40)<>32THEN400
130 POKE1946+G,95:POKE56218+G,13
135 POKE1947+G,105:POKE56219+G,13
136 POKE1948+G,32
150 IFB=2THENPOKEN,32:N=N-1:GOTO110
160 IFB=3THENPOKEN,32:N=N+1:GOTO110
170 IFB=4THENPOKEN,32:N=N+40:GOTO110
180 IFB=5THENPOKEN,32:N=N-40:GOTO110
190 IFB>5THEN210
200 GOTO110
210 P=PEEK(203)
220 IFP=39THENPOKEN,32:N=N-1:GOTO110
230 IFP=36THENPOKEN,32:N=N+1:GOTO110

```

```

240 IFP=12THENPOKEN,32:N=N+40:GOTO110
250 IFP=10THENPOKEN,32:N=N-40
260 GOTO110
300 J=J-1
301 K=K+1
302 PRINT"SRTIME"K
305 IFJ=0THEN700
306 ZX=INT(RND(1)*2)
307 IFZX=1THENDF=-1
308 IFZX=2THENDF=1
309 G=G+DF
310 GOTO111
400 PRINT"s_000000_
410 PRINTTAB(10)"O.K
420 PRINT"QRVOUS AVEZ REUSSI L'ATERRISSA
GE"
421 PRINT"QET IL RESTAIT"J"UNITES
440 PRINT"QAVEC L'ORDRE DE FACILITE"A
450 PRINT"QR UNE AUTRE PARTIE ?
454 POKE198,0
455 GETR$
460 IFR$="0"THEN0
465 IFR$="N"THENPRINT"s_":END
470 GOTO455
501 POKES3281,11:PRINT"s_000000_
502 INPUT"ORDRE DE FACILITE (5-20) ";A
503 IFA>20ORA<5THENPRINT"ERREUR !!!":GOTO
502
508 PRINT"s_000000_
511 PRINT" VOTRE VAISSEAU VIENT D'ETRE
520 PRINT"RTOUCHEZ; IL VOUS FAUT REJOIND
RE
530 PRINT" LA BASE MALGRE QUE LA DIREC
TION
540 PRINT" SOIT SERIEUSEMENT TOUCHEE
550 PRINT"Q ALORS BONNE CHANCE !!!
560hPRINT"VOUS AVEZ 200 UNITES DE CARBUR
ANT
570 PRINT"POUR REALISER L'ATERRISSAGE
580 PRINT"QAPPUYEZ SUR SHIFT POUR FAIRE
UNE PARTIE
590 WAIT653,1:GOTO1
700 PRINT"s_000000_
710 PRINT" RVOUS AVEZ PERDU !!!
720 PRINT" QVOUS VOUS ETES ECRASE
730 PRINT"Q APPUYEZ SUR SHIFT POUR REJO
UER !!!
740 WAIT653,1:RUN

```

READY.



```

4050 [FPEEK(I)<40]THENPOKEI,40
4060 [FPEEK(I+1)>165]THENPOKEI+1,165
4070 [FPEEK(I+1)<70]THENPOKEI+1,70
4080 NEXT RETURN
5000 REM PRISE DU TRESOR
5005 D=ABS(PEEK(U)-PEEK(U+8)).IFD>48[THEN
830
5006 D=ABS(PEEK(U+1)-PEEK(U+9)).[FD>42]TH
FN830
5010 FORI=1TO5
5020 POKEU+21,15:FORJ=1TO50:NEXTJ:POKEU+
21,255:FORJ=1TO50:NEXTJ:I
5030 POKEU+9,0:A=PEEK(U+30)
5035 T=T+1
5040 GOTO700
6000 REM PRESENTATION ET INSTRUCTIONS
6005 POKE53281,4
6010 PRINT"EN TAB(5)">>> monstres & tre
sors <<<
6020 PRINT" VOUS ETES UN AVENTURIER QU
I EST ENTRE
6030 PRINT" DANS UN CHATEAU PEUPLE DE MON
STRES
    
```

```

6040 PRINT"POUR Y RAFLER DES TRESORS !!!
6045 PRINT" VOUS ETES MUNI D'UN ARC MA
GIQUE DONT
6050 PRINT" LES FLECHES FONT DISPARAITRE
TOUT CE
6060 PRINT" QU'ELLES TOUCHENT.
6070 PRINT" VOICI COMMENT VOUS DIRIGER

6080 PRINT" - a - POUR ALLER AU NORD
6090 PRINT" - z - ----- AU SUD
6100 PRINT" - n - ----- A L'OUES
T
6110 PRINT" - m - ----- A L'EST
6120 PRINT" APPUYEZ SUR LA BARRE D'ESP
ACE POUR
6130 PRINT" DECOCHER UNE FLECHE MAGIQUE
6135 PRINT" VOUS POUVEZ UTILISER LE JO
YSTICK 2.
6140 PRINT" APPUYEZ SUR SHIFT POUR COM
MENCER !!
6150 WAIT653,1:RETURN

READY.
    
```

**JOURNAL LUMINEUX**

**Vous connaissez le principe pour l'avoir vu dans moult expositions. Vous tapez une phrase qui se déroule ensuite en grosses lettres sur l'écran. Tous les caractères et les couleurs sont utilisables, y compris en contraste inversé. Les majuscules et les minuscules peuvent coexister : pour avoir une majuscule, tapez-la en position mode majuscule-graphique ; pour avoir une minuscule, passez en mode minuscule par Shift C=. Pour avoir une couleur ou le contraste inversé, faites CTRL ou C= et la touche numérique voulue.**

```

10 DIM KS(254),CL$(15)
20 SC=53248:CA=1024:CR$=CHR$(13)
25 FORI=1TO8:D$=D$+"QI"+CHR$(20):NEXTI
30 FORI=0TO15:READCL$(I):NEXTI
50 REM SAISIE DU TEXTE
60 N=-1:PRINT"CHR$(142)";
70 PRINT" JOURNAL LUMINEUX"
80 PRINT:PRINT" EN GROSSES LETTRES"
90 PRINT:PRINT:PRINT" DONNEZ VOTRE TEXTE
":PRINT
95 PRINT"?";
100 NL=PEEK(214):NC=PEEK(211):NC=NC-40*IN
T(NC/40)
120 PRINT"?";
130 I=0
140 I=I+1:GETX$:IFX$<>"" THEN 200
150 IFI<50 THEN 140
160 J=1-J:IF J=0 THEN 120
170 PRINT" ";:GOTO130
200 X=ASC(X$):IFX=170RX=290RX=1450RX=157
THEN100
201 IFX=190RX=1470RX=148 THEN 100
202 IF X=13 THEN 300
205 PRINTX$:IF X=34 THEN PRINTCHR$(34)C
HR$(20);
207 IF X=20 AND N>-1 THEN KO$=LEFT$(KO$,
N):N=N-1:GOTO100
210 IF ASC(X$)<32 THEN 100
220 IF ASC(X$)>127 AND ASC(X$)<160 THEN
100
230 N=N+1:KS(N)=PEEK(CA+NL*40+NC)*8+SC+(
PEEK(53272)AND2)*1024
    
```

```

240 KO$=KO$+CL$(PEEK(646))
250 IF .NK254 THEN 100
300 REM AFFICHAGE DU TEXTE
301 PRINT"CHR$(142)";
310 FOR L=0 TO N:K1=KS(L):K2=K1+7
320 PRINTMID$(KO$,L+1,1)
330 FOR K=7 TO 0 STEP -1:X$="":PK=2+K
335 POKE56334,PEEK(56334)AND254
336 POKE1,PEEK(1)AND251
340 FOR KS=K1 TO K2
350 IF PEEK(KS)ANDPK THEN X$="q△"+X$:GO
TO370
360 X$=" q△"+X$
370 NEXT KS
375 POKE1,PEEK(1)OR4
376 POKE56334,PEEK(56334)OR1
380 PRINT"SOOOOOOO" D$TAB(38)X$;
390 NEXT K,L:GOTO310
1000 DATA"p","e","r","o","_","^","<","n"
1010 DATA"u","l","v","w","x","z","z","l"
    
```







AF14 44820 vérifie domaine  
 AF28 44840 cherche variable  
 AFA7 44967 référence à FN...  
 AFE6 45030 effectue OR  
 AFE9 45033 effectue AND  
 B016 45078 comparaison  
 B081 45185 effectue DIM  
 B08B 45195 localise variable  
 B113 45331 vérif car. alphabétique  
 B11D 45341 crée une variable  
 B194 45460 sp pointeur vers tableau  
 B1A5 45477 valeur 32768  
 B1B2 45490 convers. flottant/entier  
 B1D1 45521 prépare tableau  
 B245 45637 "BAD SUBSCRIPT"  
 B248 45640 "ILLEGAL QUANTITY"  
 B34C 45900 calcule taille tableau  
 B37D 45949 effectue FRE  
 B391 45969 conv. entier-flottant  
 B39E 45982 effectue POS  
 B3A6 45990 vérifie mode direct  
 B3B3 46003 effectue DEF  
 B3E1 46049 vérifie syntaxe de FN  
 B465 46181 effectue STR\$  
 B475 46197 calcule vect de chaîne  
 B487 46215 prépare chaîne  
 B4F4 46324 fait de la place pour  
     une chaîne  
 B526 46374 "garbage collection"  
 B5BD 46525 teste si à conserver  
 B606 46598 rassemble chaîne  
 B63D 46653 concaténation  
 B67A 46714 construit chaîne  
     en mémoire  
 B6A3 46755 rejette chaîne  
 B6DB 46811 nettoie la pile des  
     descripteurs  
 B6EC 46828 effectue CHR\$  
 B700 46848 effectue LEFT\$  
 B72C 46892 effectue RIGHT\$  
 B737 46903 effectue MID\$  
 B761 46945 dépèle les paramètres  
     de chaîne  
 B77C 46972 effectue LEN  
 B782 46978 sort du mode chaîne  
 B78B 46987 effectue ASC  
 B79B 47003 obtient octet paramètre  
 B7AD 47021 effectue VAL  
 B7EB 47083 obtient les paramètres  
     pour POKE/WAIT  
 B7F7 47095 conv. flottant-entier  
 B80D 47117 effectue PEEK  
 B824 47140 effectue POKE  
 B82D 47149 effectue WAIT  
 B849 47177 ajoute 0.5  
 B850 47184 opérande de soustraction  
 B853 47187 effectue soustraction  
 B86A 47210 effectue addition  
 B947 47431 complémente ACC1  
 B97E 47486 "OVERFLOW"

B983 47491 multiplie par octet nul  
 B9EA 47594 effectue LOG  
 BA2B 47659 effectue multiplication  
 BA59 47705 multiplication d'un bit  
 BA8C 47756 mémoire vers ACC2  
 BAB7 47799 ajuste ACC1/ACC2  
 BAD4 47828 dépassement/sous-dép  
 BAE2 47842 multiplie par 10  
 BB12 47890 effectue division  
 BBA2 48034 mémoire vers ACC1  
 BBC7 48071 ACC1 vers mémoire  
 BBFC 48124 ACC2 vers ACC1  
 BC0C 48140 ACC1 vers ACC2  
 BC1B 48155 arrondit ACC1  
 BC2B 48171 obtient le signe  
 BC39 48185 effectue SGN  
 BC58 48216 effectue ABS  
 BC5B 48219 compare ACC1 et mémoire  
 BC9B 48283 conv. flottant-entier  
 BCCC 48332 effectue INT  
 BCF3 48371 chaîne vers ACC  
 BD7E 48510 prend chiffre ASCII  
 BF16 48918 constantes décimales  
 BF3A 48954 constantes pour TI  
 BF71 49009 effectue SQR  
 BF7B 49019 effectue "puissance"  
 BFB4 49076 effectue "opposé"  
 BFED 49133 effectue EXP  
 E043 57411 évaluation de série (1)  
 E059 57433 évaluation de série (2)  
 E097 57495 effectue RND  
 E0F9 57593 points d'arrêt  
 E12A 57642 effectue SYS  
 E156 57686 effectue SAVE  
 E165 57701 effectue VERIFY  
 E168 57704 effectue LOAD  
 E1BE 57790 effectue OPEN  
 E1C7 57799 effectue CLOSE  
 E1D4 57812 paramètres pour  
     LOAD/SAVE  
 E206 57862 vérifie param par défaut  
 E20E 57870 teste virgule  
 E219 57881 paramètres pour  
     OPEN/CLOSE  
 E264 57957 effectue COS  
 E26B 57963 effectue SIN  
 E2B4 58036 effectue TAN  
 E30E 58126 effectue ATN  
 E37B 58235 reset tiède  
 E394 58260 initialisation  
 E3A2 58274 routine CHRGET (sera  
     copiée en page 0)  
 E3BF 58303 initialise BASIC  
 E447 58439 vecteurs /copiés en \$300  
 E453 58451 initialise les vecteurs  
 E45F 58463 message à la mise sous  
     tension  
 E500 58624 prépare adresses E/S  
 E505 58629 établit limites écran  
 E50A 58634 met/obtient ligne/col  
 E518 58648 initialise E/S

E544 58692 vide écran  
 E566 58726 curseur à l'origine  
 E56C 58732 établit pointeurs écran  
 E5A0 58784 prépare val par défaut des E/S  
  
 E5B4 58804 entrée depuis le clavier  
 E632 58930 entrée depuis l'écran  
 E684 59012 teste guillemet  
 E691 59025 prépare impression écran  
 E6B6 59162 avance curseur  
 E6ED 59117 recule curseur  
 E701 59137 recule dans la ligne précédente  
  
 E716 59158 sortie écran  
 E87C 59516 va à la ligne  
 E891 59597 retour chariot  
 E8A1 59553 vérifie diminution de ligne  
  
 E8B3 59571 vérifie augmentation de ligne  
  
 E8CB 59595 établit code couleur  
 E8DA 59610 table des codes couleur  
 E8EA 59726 déroulement écran  
 E965 59749 fait place sur l'écran  
 E9C8 59851 déplace ligne sur écran  
 E9E0 59872 transfert synchro couleur  
  
 E9F0 59888 établit début de ligne  
 E9FF 59903 vide ligne écran  
 EA13 59923 imprime sur écran  
 EA24 59940 synchro couleur caract.  
 EA31 59953 interruption IRQ  
 EA87 60039 teste clavier  
 EB79 60281 vecteurs clavier  
 EB81 60289 carte clavier  
 EC44 60484 controle graphique/texte  
 EC4F 60495 met en mode graphique  
 EC78 60536 matrice des touches avec CTRL  
  
 ECB9 60601 options par défaut 6567  
 ECFO 60656 adr. basse ligne écran  
 ED09 60681 envoie "parle"  
 EDOC 60684 envoie "écoute"  
 ED25 60709 envoie carac de commande  
 ED40 60736 envoi sur bus série  
 EDB2 60850 time-out sur bus série  
 EDB9 60857 envoie adresse secondaire écouteur  
  
 EDBE 60862 RAZ ATN  
 EDC7 60871 envoie adresse secondaire  
 EDCC 60876 attente horloge daire parleur  
  
 EDDD 60893 envoi série différé  
 EDEF 60911 env. cessation émission  
 EDFE 60926 envoie cessation écoute  
 EE13 60947 réception sur bus série  
 EE85 61061 met à 1 ligne horloge  
 EE8E 61070 met à 0 ligne horloge  
 EE97 61079 sortie série à 1  
 EEA0 61088 sortie série à 0  
 EEA9 61097 acquiert donnée+horloge

EEB3 61107 délai 1 ms  
 EEBB 61115 envoi sur RS232 (NMI)  
 EF06 61190 envoie nouvel oct. RS232  
 EF2E 61230 erreur: modem non prêt  
 EF31 61233 err.: pas prêt à émettre  
 EF3B 61243 désactive temporisateur  
 EF4A 61258 compte de bits  
 EF59 61273 reçoit sur RS232 (NMI)  
 EF7E 61310 prépare réception  
 EFC5 61381 err parité en réception  
 EFCA 61386 err surcharge réception  
 EFCD 61389 err rupture en réception  
 EFD0 61392 err cadrage en réception  
 EFE1 61409 fichier vers RS232  
 F017 61463 envoi vers tampon RS232  
 F04D 61517 INPUT depuis tamp RS232  
 F086 61574 GET depuis tampon RS232  
 FOA4 61604 vérif bus série au repos  
 FOBD 61629 messages  
 F12B 61739 imprime si mode direct  
 F13E 61758 lecture ...  
 F14E 61774 sur RS232  
 F157 61783 entrée  
 F199 61849 GET cassette/série/RS232  
 F16A 61898 sortie ...  
 F1DD 61917 sur cassette  
 F20E 61966 établit périph d'entrée  
 F250 62032 établit périph de sortie  
 F291 62097 CLOSE  
 F30F 62223 trouve fichier  
 F31F 62239 prépare fichier  
 F32F 62255 supprime ts les fichiers  
 F333 62259 rétablit périph standard  
 F34A 62282 ouvre fichier  
 F3D5 62421 envoie adr. secondaire  
 F409 62473 ouvre RS232  
 F49E 62622 charge programme  
 F5AF 62895 "SEARCHING"  
 F5C1 62913 imprime nom de fichier  
 F5D2 62930 "LOADING/VERIFYING"  
 F50D 62941 sauve programme  
 F68F 63119 "SAVING"  
 F69B 63131 incrémente horloge  
 F6BC 63164 note la touche lue sur PIA  
  
 F6DD 63197 prend l'heure  
 F6E4 63204 met à l'heure  
 F6ED 63213 obéit à la touche STOP  
 F6FB 63227 mess d'err sur fichiers  
 F72D 63277 trouve en-tête bande  
 F76A 63338 écrit en-tête bande  
 F7D0 63440 obtient adresse tampon  
 F7D7 63447 établit pointeurs début/fin tampon  
  
 F7EA 63466 trouve en-tête spécifique  
  
 F80D 63501 incrém. pointeur bande  
 F817 63511 "PRESS PLAY ..."  
 F82E 63534 teste état cassette  
 F838 63544 "PRESS RECORD ..."  
 F841 63553 commence lecture bande



F864 63588 commence écriture bande  
 F875 63605 code commun lect./écrit  
 F8D0 63696 vérifie arrêt bande  
 F8E2 63714 établit le timing  
 F92C 63788 lit bits (IRQ)  
 FA60 64096 range caractères  
 FB8E 64398 RAZ pointeur  
 FB97 64407 prépare nouveau caract.  
 FBA6 64422 commute bande  
 FBC8 64456 écrit données  
 FBCD 64461 écrit sur bande (IRQ)  
 FC57 64599 écrit en-tête (IRQ)  
 FC93 64659 restaure vecteurs  
 FC38 64696 établit vecteur  
 FCCA 64714 arrête moteur  
 FCD1 64721 teste pointeur  
     lecture/écriture  
 FCDB 64731 incrémente pointeur  
 FCE2 64738 entrée mise sous tension  
 FD02 64770 teste si ROM en A000  
 FD10 64784 masque ROM 8000  
 FD15 64789 rétablit noyau  
 FD1A 64794 déplace noyau  
 FD30 64816 vecteurs  
 FD50 64848 init. constantes système  
 FD9B 64923 vecteurs IRQ  
 FDA3 64931 initialise registres E/S  
 FDF9 65017 sauve nom des données  
 FE00 65024 sauve détails fichiers

FE07 65031 prend l'état  
 FE1C 65052 marque ST  
 FE21 65057 marque time-out  
 FE25 65061 cherche/établit  
     haut de mémoire  
 FE27 65063 lit haut de mémoire  
 FE2D 65069 établit haut de mémoire  
 FE34 65076 cherche/établit  
     bas de mémoire  
 FE43 65091 entrée interruption NMI  
 FE66 65126 départ à chaud  
     RESET/STOP  
 FEB6 65206 restaure IRQ et sort  
 FEBC 65212 sortie IRQ  
 FEC2 65218 table de temps RS232  
 FED6 65238 entrée RS232 (NMI)  
 FF07 65287 sortie RS232 (NMI)  
 FF43 65347 fausse IRQ  
 FF48 65352 entrée IRQ principale  
 FF81 65409 table de vecteurs dont:  
 FFC6 65478 établit canal d'entrée  
 FFC9 65481 établit canal de sortie  
 FFCC 65484 restaure canaux d'E/S  
     standard  
 FFCF 65487 entrée (INPUT)  
 FFD2 65490 sortie (PRINT)  
 FFE1 65505 teste touche STOP  
 FFE4 65508 saisie (GET)  
 FFFA 65530 vecteurs hardware.

# Annonceurs

## LE GUIDE-MICRO 85-86 PARAITRA DEBUT SEPTEMBRE 85

**Le Guide-Micro**  
**est devenu l'ouvrage de référence**  
**indispensable à tous ceux qui envisagent**  
**l'achat d'un produit informatique. Tiré à plus**  
**de 100 000 exemplaires, il comprendra :**

- les fiches détaillées des micros disponibles sur le marché
- les bancs d'essai critiques des principales machines
- le tableau des imprimantes
- 2000 adresses utiles (constructeurs, boutiques, clubs)
- les mots-clefs de l'informatique

**Le Guide-Micro reste en vente trois mois chez**  
**les marchands de journaux.**

Pensez à réserver votre  
 emplacement publicitaire  
 auprès de :  
 Marie-Christine SEZNEC  
 ou Philippe DANILOFF  
 au (1) 240 22 01

LE GUIDE DE  
**L'ORDINATEUR**  
**L'INDIVIDUEL**



## « LES ADRESSES STRATEGIQUES DU C. 64 »

HEXA	DEC	DESCRIPTION
0000	0	registre direction du port parallèle dans le 6510
0001	1	registre donnée de ce port
0002	2	inutilisé
0003-0004	3-4	vecteur pour conversion reel-entier
0005-0006	5-6	vecteur pour conversion entier-reel
0007	7	caractere recherché
0008	8	indicateur mode guillemets
0009	9	sauvegarde colonne de tabulation
000	10	0=LOAD 1=VERIFY
000B	11	pointeur dans tampon d'entree/numero d'indice
000C	12	indicateur de DIM par défaut
000D	13	type : \$FF = chaîne 00 = numerique
000E	14	type : \$80 = entier 00 = reel
000F	15	indicateur memoire mode guillemets en LIST balayage DATA
0010	16	indicateur indice / FNx
0011	17	0=INPUT. \$40 = GET \$98 = READ
0012	18	signe pour TAN/indicateur de comparaisons
0013	19	indicateur actuel d'interrogation d'entree
0014-0015	20-21	valeur entiere
0016	22	rangement temporaire du pointeur de pile
0017-0018	23-24	dernier vecteur de chaîne temporaire
0019-0021	25-33	pile pour chaînes temporaires
0022-0025	34-37	zone de pointeurs utilitaires
0026-002A	38-42	zone pour le produit dans une multiplication
002B-002C	43-44	pointeur : debut du texte BASIC
002D-002E	45-46	pointeur : debut des variables
002F-0030	47-48	pointeur : debut des tableaux
0031-0032	49-50	pointeur : fin des tableaux
0033-0034	51-52	pointeur : zone des chaînes
0035-0036	53-54	pointeur utilitaire chaînes
0037-0038	55-56	pointeur : fin de memoire
0039-003A	57-58	numero de ligne BASIC courante
003B-003C	59-60	numero de ligne BASIC precedente
003D-003E	61-62	pointeur instruction BASIC pour CONT
003F-0040	63-64	numero de ligne DATA courante
0041-0042	65-66	adresse courante de DATA
0043-0044	67-68	vecteur pour INPUT
0045-0046	69-70	nom de la variable courante
0047-0048	71-72	adresse de la variable courante
0049-004A	73-74	pointeur vers variable pour FOR...NEXT
004B-004C	75-76	sauvegarde de Y ou d'operateur ou du pointeur BASIC
004D	77	accumulateur des symboles de comparaison
004E-0053	78-83	zone de travail diverse
0054-0056	84-86	vecteur de saut pour fonctions
0057-0060	87-96	zone de travail numerique diverse
0061	97	Acc. numero 1 : exposant
0062-0065	98-101	Acc. numero 1 : mantisse
0066	102	Acc. numero 1 : signe
0067	103	pointeur vers constantes pour evaluer une serie
0068	104	debordement de Acc. numero 1
0069-006E	105-110	Acc. numero 2 : exposant mantisse signe
006F	111	signe de la comparaison Acc.1 Acc.2
0070	112	arrondi de L'Acc. numero 1
0071-0072	113-114	longueur tampon-cassette / pointeur serie
0073-008A	115-138	sous-programme CHRGET (obtient un caractere Basic)
007A-007B	122-123	pointeur Basic (dans CHRGET)
008B-008F	139-143	racine pour RND
0090	144	mot d'état ST
0091	145	copie du PIA clavier : indicateurs de STOP et RVS
0092	146	constante de vitesse pour cassette
0093	147	0 = LOAD ; 1 = VERIFY
0094	148	sortie serie : indicateur de caractere differe
0095	149	caractere serie differe
0096	150	synchro cassette
0097	151	sauvegarde de registre
0098	152	nombre de fichiers ouverts
0099	153	peripherique standard d'entree (normalement 0)
009A	154	peripherique de sortie standard determine par CMD (normalement 3)
009B	155	parite du caractere sur cassette
009C	156	indicateur de caractere reçu
009D	157	contrôle de sortie : directe = \$80 RUN = 0
009E	158	temoin d'erreur sur cassette passe 1/tampon caractere
009F	159	temoin d'erreur corrigee sur cassette (passe 2)
00A0-00A2	160-162	horloge au 60e de seconde
00A3	163	compte de bits serie / indicateur d'E01
00A4	164	compteur de cycles
00A5	165	decompteur ecriture cassette / compteur de bits
00A6	166	pointeur dans tampon cassette
00A7	167	compteur en-tete ecriture cassette / passe lecture / bit entree
00A8	168	nouvel octet ecriture cassette / erreur lecture / compte bits entrees
00A9	169	écrit bit départ / erreur lecture bit / bit de depart
00AA	170	balayage cassette / assemblage octet
00AB	171	longueur en-tete ecriture / checksum lecture / parite
00AC	172-173	pointeur : tampon cassette déroulement ecran
00AE-00AF	174-175	adresse fin cassette / fin du programme
00B0-00B1	176-177	constantes de vitesse pour cassette
00B2-00B3	178-179	pointeur : debut du tampon cassette
00B4	180	1 = temporisateur cassette active /compte de bits
00B5	181	E01 cassette / prochain bit à envoyer en RS232
00B6	182	erreur caractere en lecture / tampon octet à sortir
00B7	183	nombre de caracteres du nom de fichier
00B8	184	fichier logique courant
00B9	185	adresse secondaire courante
00BA	186	peripherique courant
00BB-00BC	187-188	pointeur vers nom de fichier
00BD	189	mot à decaler en sortie / caractere entre
00BE	190	nombre de blocs restant à lire ou ecrire
00BF	191	tampon mot serie
0297	663	etat du RS232
0298	664	nombre de bits à envoyer
0299-029A	665-666	code vitesse du RS232
029B	667	pointeur reception RS232

029C	668	pointeur entree RS232
029D	669	pointeur emission RS232
029E	670	pointeur sortie RS232
029F-02A0	671-672	sauvegarde IRQ durant E/S cassette
02A1	673	active RS232
02A2	674	horloge IR pendant operation cassette
02A3	675	donnee temporaire pendant lecture cassette
02A4	676	indicateur IRQ pendant lecture cassette
02A5	677	index de ligne temporaire
02A6	678	0=NTSC 1=PAL
02A7-02FF	679-767	inutilise
0300-0301	768-769	pointeur message d'erreur
0302-0303	770-771	pointeur vers reset tiede de Basic
0304-0305	772-773	pointeur vers routine de codage des mots-cles Basic
0306-0307	774-775	pointeur vers routine d'impression des codes des mots-cles
0308-0309	776-777	pointeur vers depart nouvelle ligne Basic
030A-030B	778-779	pointeur vers routine d'evaluation des codes BASIC
030C-030F	780-783	zone à partir de laquelle sont charges A X Y et P lors d'un SYS
0310	784	instruction de saut (\$4C) pour USR
0311-0312	785-786	pointeur vers adresse d'USR (B248)
0313	787	libre
0314-0315	788-789	vecteur interruption hardware (\$EA31)
0316-0317	790-791	vecteur de BRK (FE66)
0318-0319	792-793	vecteur de NMI (FE47)
031A-031B	794-795	vecteur pour OPEN (F34A)
031C-031D	796-797	vecteur pour CLOSE (F291)
031E-031F	798-799	vecteur pour etablir l'entree standard (F20E)
0320-0321	800-801	vecteur pour etablir la sortie standard (F250)
0322-0323	802-803	vecteur pour restaurer les E/S standard (F333)
0324-0325	804-805	vecteur pour INPUT (F157)
0326-0327	806-807	vecteur pour sortie (F1CA)
0328-0329	808-809	vecteur pour tester STOP (F6ED)
032A-032B	810-811	vecteur pour GET (F13E)
032C-032D	812-813	vecteur pour abandonner les E/S (F32F)
032E-032F	814-815	vecteur pour USR (FE66)
0330-0331	816-817	pointeur pour LOAD (F4A5)
0332-0333	818-819	pointeur pour SAVE (F5ED)
033C-033D	820-821	tampon cassette
0400-07FF	1024-2047	memoire d'ecran
0800-9FFF	2048-40959	texte BASIC
8000-9FFF	32768-40959	(option) ROM en cartouche
A000-BFFF	40960-49151	RAM ou ROM BASIC
C000-CFFF	49152-53247	RAM
D000-D02E	53248-53294	boitier V.I.C. 6567
D400-D41C	54272-54300	boitier SID 6581 (synthetiseur)
D800-DBFF	55296-56319	RAM couleur (quartets)
DC00-DD0F	56320-56335	6526 n°1
DD00-DDFF	56376-56591	6526 n°2
E000-EFFF	53248-57343	generateur de caracteres
E000-EFFF	57344-65535	RAM ou ROM noyau du S.E.
00C0	192	commande moteur cassette
00C1-00C2	193-194	adresse depart d'E/S
00C3-00C4	195-196	pointeur routine de preparation du noyau
00C5	197	derniere touche appuiee
00C6	198	nombre de caracteres dans le tampon clavier
00C7	199	indicateur de contraste inverse
00C8	200	pointeur vers fin de ligne en entree
00C9-00CA	201-202	temoin curseur en entree (ligne/colonne)
00CB	203	touche enfoncee : 64 si aucune
00CC	204	0 = le curseur clignote
00CD	205	decompteur pour vitesse de clignotement curseur
00CE	206	caractere sous le curseur
00CF	207	phase de clignotement curseur
00D0	208	entree depuis l'ecran ou le clavier
00D1-00D2	209-210	pointeur vers ligne ecran
00D3	211	position du curseur dans la ligne
00D4	212	0 = curseur direct : sinon programme
00D5	213	longueur ligne courante sur ecran
00D6	214	ligne ou se trouve le curseur
00D7	215	tampon / checksum / derniere touche
00D8	216	nombre d'insertions en attente
00D9-00FD	217-240	table des liens des lignes-ecran
00F1	241	lien bidon /
00F2	242	marqueur ligne ecran
00F3-00F4	243-244	pointeur memoire couleur ecran
00F5-00F6	245-246	pointeur table decodage clavier
00F7-00F8	247-248	pointeur vers reception RS232
00F9-00FA	249-250	pointeur vers emission RS232
00FB-00FE	251-254	libre
00FF	255	donnee temporaire BASIC
0100-01FF	256-511	pile processeur dont sont utilises temporairement :
00FF-010A	255-266	zone de travail pour la conversion reel ASCII
0100-013E	256-318	memorisation erreurs cassette
0200-0258	512-600	tampon d'entree du Basic
0259-0262	601-610	table des fichiers logiques
0263-026C	611-620	table des numeros de peripheriques
026D-0276	621-630	table des adresses secondaires
0277-0280	631-640	tampon clavier
0281-0282	641-642	plus petite adresse de RAM Basic
0283-0284	643-644	plus grande adresse de RAM Basic
0285	645	indicateur de time-out sur le bus serie
0286	646	code couleur courant
0287	647	couleur sous le curseur
0288	648	page de la memoire d'ecran
0289	649	taille maximum du tampon d'ecran (doit être < 10)
028A	650	repetition automatique des touches : 0 = touches
028B	651	curseur : 127 = aucune touche 128=toutes les touches
028C	652	compteur pour la vitesse de repetition
028D	653	compteur de delai pour la repetition
		indicateur de SHIFT C= CTRL : 0=rien 1=SHIFT
		2=C= 4= CTRL
028E	654	dernier motif de SHIFT
028F-0290	655-656	pointeur vers la routine d'etablissement de la table de transcodage clavier
0291	657	mode des touches \$00: desactive les touches "SHIFT"; \$80: Les active
0292	658	0 = déroulement ecran autorise
0293	659	contrôle du RS232
0294	660	commande du RS232
0295-0296	661-662	temps d'un bit pour le RS232

NOTE : ces adresses strategiques et routines systemes ont été publiees dans la Commode, numeros 8 et 9.



# TRUCASTUCES

## LA FOIRE AUX PROGRAMMES

**Mémoire, vidéo, son, Basic, clavier, magnétophone, disquettes, périphériques, com, langages... Dans le désordre, tous les « Trucastuces » de l'Ordinateur Individuel, classés pour s'y retrouver !**

### MEMOIRE

*Basic déménagement* - DE HERVE LEMARCHAND  
(copie de Basic en MEV) L'OI n° 63 p. 225

*Bonjour programme* - DE MICHEL PIPERAN, HERVE LEMARCHAND  
(40K de MEV avec les cart. UNNEW)  
L'OI n° 64 p. 249

### VIDEO

*Curseur clignotant* - DE JEAN-PIERRE LALEVEE  
(curseur pendant un get) L'OI n° 56 p. 217

*Curseur fantôme* - DE JEAN-PIERRE LALEVEE  
(suppression du curseur en INPUT)  
L'OI n° 61 p. 200

*Programme vu, pas vu* - DE JEAN-PIERRE LALEVEE  
(tabulation verticale) L'OI n° 56 p. 217

*A la bonne ligne* - DE CHRISTOPHE THOMAS  
(tabulation verticale) L'OI n° 65 p. 251

*Sous la main* - DE PASCAL MEURISSE  
(PRINT AT (x, y)) L'OI n° 62 p. 223

### CLAVIER

*Un mot pour un autre* - DE FRANCK BONBLED  
(programmation des touches de fonction)  
L'OI n° 62 p. 223

*Menu « A la carte »* - DE LAURENT NICHON  
(test touches de fonction) L'OI n° 67 p. 163

*Un clavier AZERTY en vaut deux* - DE HERVE LEMARCHAND  
(clavier Azerty) L'OI n° 64 p. 249

*REP, REP, répète* - DE J.-F. BRIOUX  
(répétition de touches) L'OI n° 65 p. 251

*Sous la main* - DE PASCAL MEURISSE  
(répétition de touches) L'OI n° 62 p. 223

*INPUT de choc* - DE P. PARRO  
(INPUT même les virgules) L'OI n° 65 p. 251

*Répétition sur commande* - DE PASCAL LEJEUNE  
(sensibilité répét. touche) L'OI n° 63 p. 225

*Astuces clavier* - DE JEAN-PIERRE LALEVEE  
(17 astuces clavier) L'OI n° 57 p. 000

*Sous la main* - DE PASCAL MEURISSE  
(attente de frappe) L'OI n° 62 p. 223

### MAGNETOPHONE

*A vos cassettes* - DE JEAN-PIERRE LALEVEE  
(commander le magnét. par PRG)  
L'OI n° 61 p. 200

*Erreur de lecture* - DE JEAN-PIERRE LALEVEE  
(tout n'est pas perdu...) L'OI n° 61 p. 197

### DISQUETTES

*Fichier accidenté* - DE RALF WINZER  
(rouvrir un fichier mal fermé) L'OI n° 66 p. 182

*Astuces périphériques* - DE HERVE LEMARCHAND  
(adresse début PRG sur disque) L'OI n° 62 p. 223

*Corrigez vos disquettes* - DE JEAN-PIERRE LALEVEE  
(modifier titre et id.) L'OI n° 61 p. 197

*Sauvegarde de MEV* - DE NILS DECROP  
(sauvegarde portion MEV) L'OI n° 68 p. 186

### PERIPHERIQUES

*Astuces périphériques* - DE HERVE LEMARCHAND  
(Hardcopie d'écran) L'OI n° 62 p. 223

*Pour lire les directions* - DE THIERRY PAULY  
(Demo Joystick) L'OI n° 60 p. 213

### LANGAGES

*A propos de similitudes* - DE JEAN-PIERRE LALEVEE  
(adresses Vic & adresses 64) L'OI n° 65 p. 250

*Recherche dans les mémoires* - DE JEAN-PIERRE LAURENT  
(conversion CBM/64) L'OI n° 57 p. 243

*Périphériques en tous genres* - DE JEAN-PIERRE LALEVEE  
(lire sur CBM PRGS 64) L'OI n° 56 p. 217

*Alea jacta est* - DE JEAN-PIERRE LALEVEE  
(nombres aléatoires) L'OI n° 61 p. 200

*Logicomanie* - DE LAURENT NICHON  
(vrai = -1 : faux = 0) L'OI n° 65 p. 250

*Commentaires sans REM* - DE FRANCIS DURAND  
(remarque sans REM) L'OI n° 55 p. 247

*Horloge parlante* - DE G. BIDI  
(horloge LM sur écran) L'OI n° 66 p. 182

### SAUEGARDE

*Pause resto* - DE MICHAEL DUBOIS  
(inhib. STOP/RESTORE + chang. coul.)  
L'OI n° 67 p. 163

*Programme vu, pas vu* - DE JEAN-PIERRE LALEVEE  
(inhiber SAVE et LIST) L'OI n° 56 p. 217

*Taupe secrète* - DE P. SERVY  
(truquer les REMs) L'OI n° 65 p. 250

*Pas vu, pas pris* - DE CHRISTOPHE THOMAS  
(truquer les REMs) L'OI n° 65 p. 251

# LIVRES

## QUE SAIS-JE

**Le Commodore 64 a inspiré les auteurs de tous poils et les adorateurs de lutins lui ont rendu hommage en programmes fleuris. Les fouineurs ont disséqué ses puces et leurs pattes. Les professionnels de la déclinaison n'ont jamais négligé CBM. Conséquence : la bibliothèque consacrée à l'appareil s'en trouve riche, vaste et variée ; nous avons opéré une sélection des ouvrages les plus récents.**

### PROGRAMMES DE JEUX

Les 102 programmes (1) sont nombreux, simples, expliqués ; ils ne dépassent jamais la page d'instructions et sont rangés par difficultés croissantes.

Les jeux à l'affiche (2) et en famille (3) sont moins nombreux, plus riches et plus complexes tout en restant diversifiés. *Faites vos jeux* (4) propose au contraire de gros programmes, parfois très originaux, soigne avec succès le graphisme et le son, explique peu et POKE beaucoup. Les 18 jeux en Basic (5), tous des classiques bien présentés mais non expliqués, bénéficient de leur coût modeste. Les *Jeux sur Commodore* (deux volumes) (6) sont peu nombreux, onéreux mais soignés et bien expliqués, dans la conception et les techniques de programmation, autant que ligne à ligne. Les 21 jeux sur le Commodore 64 (7) offrent de l'éclectisme, une certaine originalité et une bonne pré-

sentation ; à leur charge, des explications de programmes sommaires et un prix élevé. *Le livre de jeux* (8) se compose de dix-huit très beaux programmes qui profitent fort bien des richesses graphiques et sonores de l'appareil ; les quelques jeux originaux valent les autres, plus classiques ; tous sont expliqués dans un détail accessible au « Basiqueur » commun. (La couverture est superbe ; enfin, j'aime beaucoup.) Les 29 jeux d'arcade (9), classiques, très joliment présentés, n'hésitent pas à faire appel au langage machine pour accélérer leur déroulement ; ils nous offrent des représentations d'écran en couleur sur papier glacé, c'est cher mais c'est beau. Les *Jeux pour apprendre à programmer* (10) saucissonnent douze jeux en « modules » bien expliqués ; ils PEEK et POKE lestement et ouvrent de véritables mines à ceux qui veulent progresser en program-

### UTILITAIRES

Une vingtaine de programmes, tous utilitaires, composent les *Activités* (12) ; graphisme, musique, traitement de texte ou fichiers en font l'objet.

Les autres livres que nous avons rangés au chapitre « utilitaires » proposent un choix éclectique de programmes, allant du jeu à l'utile en passant par le fonctionnel et l'amusant ; tous conviennent à la découverte d'un usage général ou particulier

du C. 64.

Les 30 programmes (13) – maths, jeux, astronomie... –, bien que très discrètement explicités, méritent une bonne mention pour leur rapport prix/diversité. La trentaine de *Programmes en Basic* (14) explore correctement le C. 64 mais n'étouffent pas le lecteur sous les commentaires. Les *Méthodes pratiques* (15) mêlent exercices et méthodes pour aider l'opérateur à fouiller les

### POUR ALLER PLUS LOIN

La pratique (17) est, en gros, un mode d'emploi approfondi des périphériques. Les *Clefs pour le commodore 64* (18) sont un mémento de références des codes, syntaxes, messages, adresses, etc., utiles de l'appareil. Les *Trucs et astuces* (19) sont destinés aux programmes avancés qui y trouveront une multitude d'idées de routines et de ficelles en tous genres ; tout comme *PEEKs ET POKES* (20) du même éditeur, qui décrit et explique les voies de pénétration dans les profondeurs du C. 64 ; sans oublier *Le livre de l'imprimante* (21), *Le livre du lecteur de disquette 1541* (22), ou du *Lecteur de cassette 1530* (23), *Le langage machine* (24) ou *L'anatomie du 64* (25), qui livrent le listing complet de la ROM et les schémas originaux de la machine. La collection de Micro Application n'est ni simple (loin s'en faut), ni bon mar-

ché (ce n'est pas original), mais fournit à l'utilisateur chevronné tous les éléments dont il peut avoir usage. Les amateurs d'assembleur sont servis, on l'aura compris ; le second tome de *La pratique* (27) s'intéresse au jeu d'instructions du 6502 (6510) et au langage machine ; le *Programme interne du Commodore 64* (28) est un listing aussi sec que ça des 16 K de ROM du Commodore 64. Ceux qui arrivent jusque-là n'auront plus grand-chose à apprendre de leur machine. Enfin, et pour la bonne bouche, le *Programmer USER\$'s guide*, et le *Livre du C. 64* (31) de Benoît Michel, in-dispensables (30).

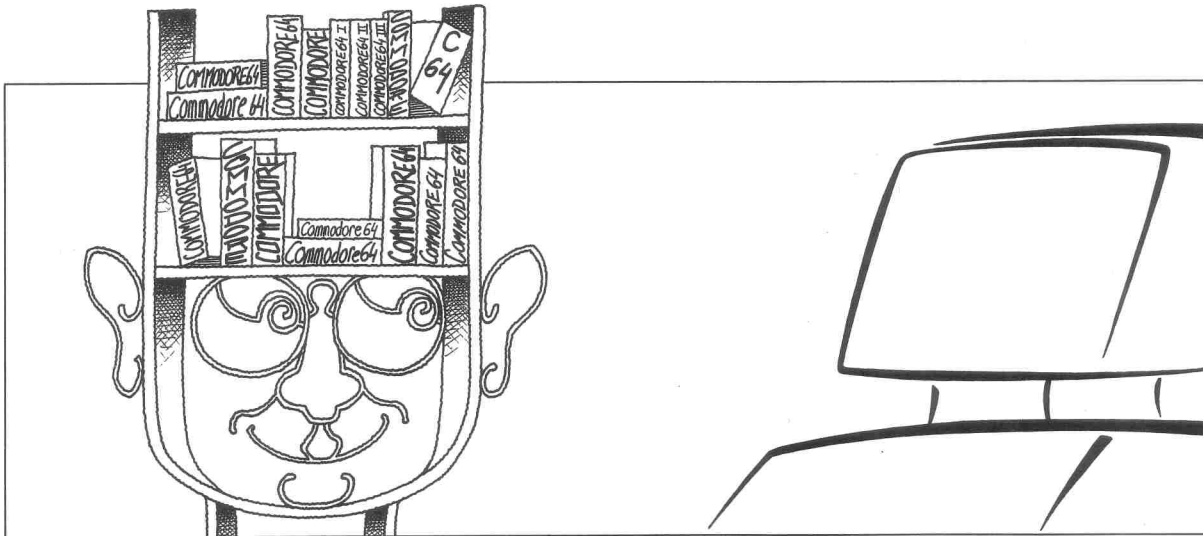
#### 17. La pratique du Commodore 64

D.-J. David - PSI - 176 pages - 95 FF.

#### 18. Clefs pour le Commodore 64

D.-J. David - PSI - 126 pages - 105 FF.





mation Basic. *Destination aventure* (11) offre une authentique leçon de programmation pour jeux de rôles et d'aventure ; on y trouve quatre grands programmes, mais surtout une foule de trucs pour inventer soi-même, et une disquette pour ne pas devoir (se) taper tous les listings.

**1. 102 programmes pour Commodore 64**

J. Deconchat - PSI - 240 pages - 110 FF.

**2. Commodore 64 à l'affiche**

J.-F. Sehan - PSI - 176 pages - 100 FF

**3. Commodore 64 en famille**

J.-F. Sehan - PSI - 224 pages - 110 FF.

**4. Faites vos jeux avec Commodore 64**

M. Ducamp et P. Schaffer - Microplus - 192 pages - 90 FF.

**5. Jeux en Basic sur Commodore 64**

P. Monsaut - Sybex - 90 pages - 49 FF.

**6. Jeux sur Commodore 64**

(1 : Adresse et hasard. 2 : Action et réflexion)

P. Fleurier et C. Meiller

Edimicro Tome 1 : 160 pages - 88 FF.

Tome 2 : 152 pages - 88 FF.

**7. Jeux sur le Commodore 64**

O. Bishop - Belin - 126 pages - 95 FF.

**8. Le livre de jeux du Commodore 64**

E. Massé et N. Chatelain - Eyrolles - 150 pages - 90 FF.

**9. Les jeux d'Arcade sur Commodore 64**

C. et M. Ramshaw - Mémoire vive - PSI - 196 pages - 120 FF.

**10. Jeux pour apprendre à programmer**

F. Gangloff - Eyrolles - 124 pages - 82 FF.

**11. Destination aventure**

D.T. Horn - PSI - 244 pages - 140 FF.

D.T. Horn - Disquette - 210 FF.

fonctions de sa machine. Les *Jeux, trucs et comptes* (16) entremêlent jeux, gestion et ficelles en trente programmes pour satisfaire les vices variés du programmeur motivé. Celui qui penche vers le *Graphisme* (26) ou la *Musique* (29) trouvera la matière à satisfaire sa curiosité.

**12. Activités avec le Commodore 64**

D. Lasseran - Hachette Informatique - 189 pages - 95 FF.

**13. 30 programmes pour Commodore 64**

D. Lasseran - ETSF - Poche informatique - 125 pages - 35 FF.

**14. Programmes en Basic pour Commodore 64**

T. Knight-D.Labatt - Modulo Belin - 115 pages - 95 FF.

**15. Commodore 64 : méthodes pratiques**

J. Boisgontier - PSI - 176 pages - 105 FF.

**16. Jeux, trucs et comptes pour Commodore 64**

M. Benelfoul-C. Cambien - PSI - 192 pages - 120 FF.

**19. Trucs et astuces du Commodore 64**

Angerhausen - English - Gerits - Micro Application - 230 pages - 149 FF.

**20. PEEKS et POKES**

Liesert - Micro Application - 196 pages - 99 FF.

**21. Le livre de l'imprimante Commodore 64 et VIC 20**

Brückmann-Weins-Gerits - Micro Application - 280 pages - 179 FF.

**22. Le livre du lecteur de disquette 1541**

Lothar-English-Norbert - Micro Application - Szczepanowski - 300 pages - 179 FF.

**23. Le livre du lecteur de cassette 1530**

Paulissen - Micro Application - 160 pages - 99 FF.

**24. Le langage machine du Commodore 64 - Tome 1**

Lothar-English Micro Application - 255 pages - 149 FF.

**25. L'anatomie du C. 64**

Lothar-English-Norbert - Micro Application-Szczepanowski - 300 pages - 199 FF.

**26. Le graphisme du Commodore 64**

Axel Plenge - Micro Application - 295 pages - 149 FF.

**27. L'assembleur du Commodore 64 (pratique - tome 2)**

D.-J. David - PSI - 208 pages - 105 FF.

**28. Programme interne du Commodore 64**

M. Bathurst - Datacap - PSI - 252 pages - 140 FF.

**29. Musique sur Commodore 64**

J. Vogel et N.B. Scrimshaw - Cedic/Nathan - 159 pages - 85 FF.

**30. Programmer User's Guide**

C.B.M. - 490 pages - 160 FF.

**31. Le livre du C. 64**

Benoît Michel - PSI - 300 pages - 120 FF.

# AVEZ-VOUS TOUS LES LIVRES P.S.I POUR VOTRE COMMODORE ?

## POUR ETRE PLUS VITE EFFICACE AVEC VOTRE C 64

### LA DECOUVERTE DU COMMODORE 64

Un apprentissage progressif du langage Basic et les points forts du Commodore 64  
176 pages - 95,00 FF

• Par Daniel-Jean DAVID

### COMMODORE 64 POUR TOUS

L'initiation par l'exemple au Commodore 64  
176 pages - 105,00 FF

• Par Jacques BOISGONTIER,  
Sophie BREBION  
et Gérard FOUCAULT

### 102 PROGRAMMES POUR COMMODORE 64

L'exploration du Commodore 64 à travers des programmes de jeux  
240 pages - 120,00 FF

• Par Jacques DECONCHAT

### LE LIVRE DE BORD DU COMMODORE 64

Introduction au Commodore 64 et particulièrement au graphisme  
256 pages - 130,00 FF

• Par Mathieu KOKINSKI

Sans aucune connaissance préalable, vous pouvez aborder cet ouvrage dans lequel sont traitées les applications sérieuses mais aussi ludiques de l'ordinateur individuel Commodore 64. Une introduction générale sur l'informatique, suivie d'un apprentissage progressif du langage Basic vous amène pas à pas à construire vos propres programmes. Des notions nouvelles sont introduites graduellement et, grâce à elles, vous apprendrez vite à maîtriser les points forts du Commodore : les graphiques, les sons, les couleurs, la haute résolution et les "sprites".

"Commodore 64 pour tous" initie le lecteur au langage Basic du Commodore 64. Les auteurs commencent par présenter les notions fondamentales de la programmation (variables, tests, boucles...). Sont ensuite étudiées les caractéristiques intéressantes du Commodore 64: les graphiques, les sons et les sprites avec de nombreux exemples illustrés et des programmes directement commentés. Le livre une fois terminé, le lecteur pourra aisément écrire ses propres programmes.

Apprendre en se distrayant, tel est l'objectif de ce livre. Au fil de ces 102 programmes de jeux il vous guidera dans l'exploration du Basic du Commodore 64. Les programmes sont classés par niveau, chacun d'eux faisant appel à de nouvelles connaissances et à une plus grande maîtrise du Basic. Chaque niveau commence par une présentation concise de nouvelles instructions utilisées. Tous les jeux sont décrits, et les programmes abondamment commentés; un exemple d'exécution est fourni pour chaque version.

Ce livre est une introduction à l'ordinateur individuel Commodore 64, à son équipement, à ses périphériques, ainsi qu'à son langage Basic. Sont aussi abordés: la programmation simple, les possibilités graphiques et musicales, l'utilisation des fichiers, les programmes de tri, etc. L'ouvrage est complété de programmes variés, du jeu le plus simple à un mini traitement de textes, en passant par un petit programme de gestion de fichiers.

## LE BASIC EN DOUCEUR SUR COMMODORE 64

Le livre du débutant, avec exercices corrigés  
256 pages - 140,00 FF

• Par Laurian PICARD

## LES FICHIERS SEQUENTIELS EN BASIC SUR COMMODORE 64

Initiation au Basic et aux fichiers séquentiels  
160 pages - 120,00 FF

• Par Pierre FRASER

## LUI FOURNIR DES PROGRAMMES TOUT FAITS

### JEUX, TRUCS, ET COMPTES POUR COMMODORE 64

30 programmes pour jouer avec le Commodore 64  
192 pages - 120,00 FF

•• Par Michel BENELFOUL  
et Cyril CAMBIEN

### COMMODORE 64 EN FAMILLE

40 utilisations du Commodore 64 dans la vie quotidienne  
244 pages - 120,00 FF

•• Par Jean-François SEHAN

Grâce à une démarche très pédagogique, cet ouvrage initie le jeune lecteur au langage Basic du Commodore 64.

Après quelques explications préliminaires on passe directement à l'étude des instructions et des commandes: commandes de couleur, les sprites, les variables, les boucles, la programmation du son et de la musique, les sous-programmes, les tableaux, etc. Toutes ces notions sont introduites progressivement et mises en pratique dans de nombreux exercices.

Le Commodore 64 est un ordinateur individuel qui dispose d'importantes possibilités d'utilisation de fichiers. Pierre Fraser, professeur à l'université de Québec, travaille sur les ordinateurs Commodore depuis plusieurs années. Il nous présente ici une introduction à la structure et à l'utilisation des fichiers séquentiels en Basic.

Que programmer? Comment? Pour quoi faire?... "Jeux, trucs et comptes pour Commodore 64" propose aux novices de la programmation 30 programmes en Basic commentés et décrits à l'aide d'un exemple d'exécution et d'un organigramme. Fonctions Basic, jeux passifs et interactifs, interludes, "trucs", paie, facturation simple et routines, l'ensemble de ce recueil très varié, leur permettra d'utiliser au mieux toutes les fonctions de leur Commodore 64.

Que faire d'un ordinateur Commodore 64 chez soi? Peut-on gérer son compte bancaire? Prévoir le coût de sa voiture chaque année? Apprendre à compter à ses enfants? Calculer son signe astrologique chinois? C'est en réponse à ces questions que Jean-François Sehan propose une sélection de 40 programmes à caractère familial. Ce livre comporte huit parties: les finances, la pédagogie, la cuisine, les jeux nationaux, le temps, la santé, le bricolage et, pour finir, quelques utilitaires. Dans un but pédagogique, chaque programme est accompagné d'un organigramme, d'une liste des variables et d'une explication de chaque ligne Basic pour mieux assimiler les techniques de programmation ou les adapter à d'autres ordinateurs.

• accessible à tous  
•• et ••• accessibles aux initiés



## DESTINATION AVENTURE SUR COMMODORE 64

Des jeux de rôles et  
d'aventure  
248 pages - 140,00 FF

••  
Par Delton T. HORN

## LE COMMODORE 64 A L'AFFICHE

30 programmes de jeux  
avec organigrammes et  
commentaires  
184 pages - 105,00 FF

•• et ••  
Par Jean-François SEHAN

## BOITE A OUTILS POUR COMMODORE 64

De nombreux petits  
programmes pratiques prêts  
à fonctionner  
128 pages - 39,00 FF

••  
Par Cyril CAMBIEN

## LES JEUX D'ARCADE DU COMMODORE 64

29 jeux d'arcade pour  
utiliser au maximum les  
capacités graphiques et  
sonores du Commodore 64  
200 pages - 130,00 FF

••  
Par Clifford  
et Mark RAMSHAW

## POUR ALLER PLUS LOIN

### LA PRATIQUE DU COMMODORE 64

L'utilisation des  
périphériques  
176 pages - 95,00 FF

••  
Par Daniel-Jean DAVID

### COMMODORE 64 : METHODES PRATIQUES

Pour perfectionner sa  
connaissance du langage  
Basic du Commodore 64  
176 pages - 105,00 FF

••  
Par Jacques BOISGONTIER

### MULTIPLAN SUR COMMODORE 64

L'utilisation du célèbre  
tableur avec de nombreux  
modèles pratiques  
208 pages - 110,00 FF

••• et •••  
Par Hervé THIRIEZ

Il n'est pas nécessaire d'être un crack  
en informatique pour créer ses propres  
jeux ; il suffit de connaître un peu  
le langage Basic et de lire "Destination  
Aventure sur Commodore 64".  
Cet ouvrage propose au possesseur  
d'un Commodore 64 quatre programmes  
de jeux d'aventure avec  
leur mode d'emploi : ce sont la Flûte  
d'Or, la Grande Evasion, Mars et la  
Chasse au Trésor. Le lecteur pourra  
utiliser toutes les instructions graphi-  
ques, les explications et les astuces  
de programmation pour élaborer par  
la suite ses propres programmes.

Une sélection de 30 programmes de  
jeux utilisant les possibilités graphi-  
ques et sonores du Commodore 64.  
Chaque programme est accompagné  
d'un organigramme, d'une liste de va-  
riables et d'une explication de cha-  
que ligne Basic pour une adaptation  
éventuelle à d'autres ordinateurs.

Une série de petits programmes les  
plus divers : jeux, mathématiques,  
éducation, programmes utilitaires,  
accompagnés d'explications simples  
qui indiquent comment utiliser et  
comprendre les programmes (rou-  
lette russe, poursuite, équations,  
etc.).

C'est dans les jeux que les possibilités  
du Commodore 64 se réalisent le  
mieux. Voici donc un recueil de 29  
jeux, écrits en Basic et faciles à entrer  
à partir du clavier. Tous les jeux (pi-  
lotage et atterrissage d'un avion, atta-  
que des Martiens, grand slalom, etc.)  
sont expliqués en détail. Le lecteur y  
trouvera en plus un programme de  
détection des erreurs de frappe, un  
autre pour créer des sprites et une  
routine de conversion pour manches  
à balai.

De la cassette au disque souple, de  
l'imprimante aux poignées de jeux et  
crayon lumineux, ce livre vous don-  
nera tout ce qu'il faut connaître pour  
utiliser au mieux les périphériques de  
votre Commodore 64. Il contient de  
nombreux programmes écrits tant  
pour les applications personnelles  
que professionnelles. Un chapitre est  
consacré aux notions sur les bases de  
données et au système d'exploitation  
disque. La programmation de l'inter-  
face RS 232 est décrite.

Cet ouvrage contient 50 exercices et  
programmes de jeux, d'éducation et  
de gestion. Il s'adresse aux utilisate-  
urs du Commodore 64 qui, ayant  
déjà pratiqué le langage Basic, veu-  
lent en savoir plus sur les possibilités  
de cet ordinateur, en particulier sur  
les graphiques haute et basse résolu-  
tion, les sons, les sprites, les fichiers  
séquentiels et relatifs, la redéfinition  
des caractères, etc.

Multiplan est un logiciel qui permet  
de créer et de gérer plusieurs ta-  
bleaux différents simultanément. Cet  
ouvrage guide pas à pas dans l'utilisa-  
tion de Multiplan sur Commodore 64  
à l'aide d'exemples progressifs. De  
nombreux cas d'application sont pré-  
sentés : gestion de portefeuille, ges-  
tion de copropriété, de feuilles de pa-  
ye, de feuilles d'impôt, etc. Chaque  
exemple sert à introduire de nouvel-  
les instructions ou astuces d'utilisa-  
tion.

## CLEFS POUR LE COMMODORE 64

La ROM désassemblée pour  
mieux programmer en  
langage machine  
126 pages - 105,00 FF

•••  
Par Daniel-Jean DAVID

## POUR LES PROS

### BASIC +

80 routines de simulation  
pour Commodore 64  
136 pages - 85,00 FF

•••  
Par Michel MARTIN

### LE LIVRE DU 64

Tout sur le système du  
Commodore 64  
202 pages - 130,00 FF

•••  
Par Benoît MICHEL

### L'ASSEMBLEUR DU COMMODORE 64

Tout sur le jeu d'instructions  
du microprocesseur du 6510  
208 pages - 105,00 FF

•••  
Par Daniel-Jean DAVID

### PROGRAMME INTERNE DU COMMODORE 64

Toutes les adresses des ROM  
252 pages - 140,00 FF

•••  
Par Milton B. BATHURST

### PROGRAMMES INTERNE DU LECTEUR DE DISQUETTES COMMODORE 1541

Toutes les adresses du  
lecteur de disquettes 1541  
252 pages - 140,00 FF

•••  
Par Milton B. BATHURST

De la collection Mémento, cet ouvra-  
ge, consacré au Commodore 64, con-  
tient toutes les informations de réfé-  
rence à retrouver rapidement au mo-  
ment de l'utilisation de la machine :  
syntaxe des commandes, codes ca-  
ractères, messages d'erreur, langage  
machine (6502/6510), connecteur,  
adresses utiles. Pour terminer un re-  
cueil d'astuces permettra au lecteur  
d'avoir à portée de la main des "trucs"  
très précieux à une bonne utilisation  
de l'appareil.

"Basic +" propose 80 routines pour  
"muscler" l'ordinateur. Ces routines  
de simulation permettent, par exem-  
ple, sur Commodore 64 de reproduire  
le galop d'un cheval ou de jouer  
d'un instrument de musique. Les mo-  
des graphiques haute résolution don-  
nent accès aux instructions DRAW,  
CIRCLE et BOX, ainsi qu'à la pro-  
grammation des lutins (sprites), tan-  
dis que des routines d'animation gra-  
phique livrent le secret des dessins  
animés.

Avec le livre du 64 le lecteur pourra  
comprendre le fonctionnement inter-  
ne de son Commodore 64, transfor-  
mer le clavier en AZERTY accentué et  
le Basic en français utiliser différents  
modes graphiques en même temps à  
l'écran, raccorder un bouton de RE-  
SET ou une imprimante parallèle, etc.  
Très nombreux programmes d'exem-  
ples en Basic et en langage machine.

Les utilisateurs du Commodore 64  
trouveront dans cet ouvrage l'expli-  
cation et l'utilisation du jeu d'instruc-  
tion du langage machine de leur ordi-  
nateur. L'assembleur symbolique,  
l'éditeur et le chargeur y sont décrits.  
L'interaction avec Basic et avec le  
système y est étudiée. Enfin, l'oppor-  
tunité de la programmation en lan-  
gage machine est discutée (place mé-  
moire et temps d'exécution).

Ce livre est destiné à ceux qui dési-  
rent savoir comment leur Commodo-  
re 64 accomplit son travail et com-  
ment la mémoire morte (16 K) con-  
trôle toutes les opérations de leur or-  
dinateur. Le lecteur y trouvera la liste  
complète du programme interne détaillé  
et commenté avec en plus une réfé-  
rence croisée sur l'utilisation des  
variables et des routines. Cet ouvrage  
sera tout spécialement indispensable  
à ceux qui travaillent en langage ma-  
chine.

Un ouvrage indispensable pour le  
programmeur qui travaille en lan-  
gage machine avec le lecteur de dis-  
quette C1541.  
En effet ce lecteur est un véritable  
ordinateur avec microprocesseur 6502,  
16 K de mémoire morte pour son pro-  
gramme et 2 K de mémoire vive. Cet  
ouvrage en détaille toutes les capaci-  
tés et permet de savoir comment le  
programme du 1541 accomplit ses  
différentes fonctions (par exemple,  
trouver le fichier demandé, ou trans-  
mettre la table des matières de la dis-  
quette).



P.S.I. DIFFUSION B.P. 86  
77402 LAGNY-SUR-MARNE CEDEX



# COMMODORE 64

## A LIVRE OUVERT

**PLUS DE 20 LIVRES  
POUR  
VOTRE COMMODORE**  
Chez votre libraire ou en boutique spécialisée



P.S.I. : la gamme la plus complète de livres pour votre Commodore 64; de l'initiation à la performance, P.S.I. vous guide à travers les possibilités multiples de votre micro-ordinateur, en vous permettant ainsi d'en tirer le meilleur parti.

Vous trouverez dans ce numéro le détail de tous les ouvrages de P.S.I. traitant du C 64.



P.S.I. DIFFUSION B.P. 86  
77402 LAGNY-SUR-MARNE CEDEX  
Tél. : (6) 006.44.35  
Télex : PSIDIF 600 978 F